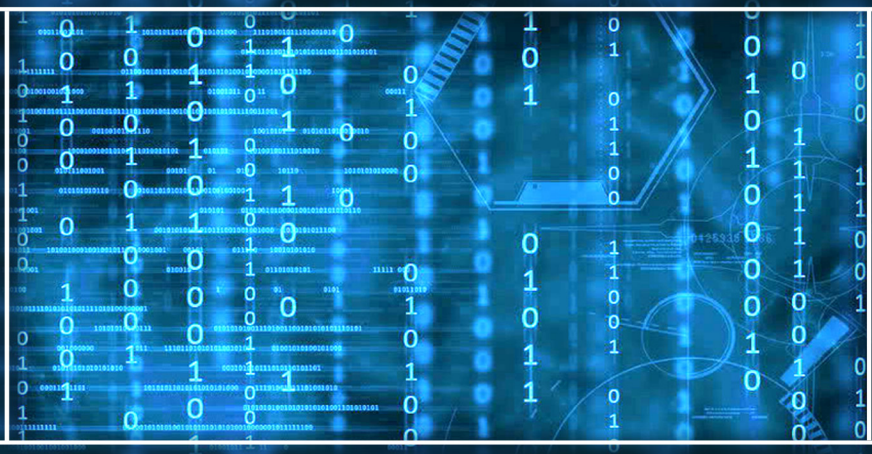
IJACSA
WHERE WISDOM SHARES
International Journal of Advanced Computer Science and Applications

Volume 16 Issue 8
August 2025



ISSN 2156-5570(Online)
ISSN 2158-107X(Print)

www.ijacsa.thesai.org

Editorial Preface

From the Desk of Managing Editor…

It may be difficult to imagine that almost half a century ago we used computers far less sophisticated than current home desktop computers to put a man on the moon. In that 50 year span, the field of computer science has exploded.

Computer science has opened new avenues for thought and experimentation. What began as a way to simplify the calculation process has given birth to technology once only imagined by the human mind. The ability to communicate and share ideas even though collaborators are half a world away and exploration of not just the stars above but the internal workings of the human genome are some of the ways that this field has moved at an exponential pace.

At the International Journal of Advanced Computer Science and Applications it is our mission to provide an outlet for quality research. We want to promote universal access and opportunities for the international scientific community to share and disseminate scientific and technical information.

We believe in spreading knowledge of computer science and its applications to all classes of audiences. That is why we deliver up-to-date, authoritative coverage and offer open access of all our articles. Our archives have served as a place to provoke philosophical, theoretical, and empirical ideas from some of the finest minds in the field.

We utilize the talents and experience of editor and reviewers working at Universities and Institutions from around the world. We would like to express our gratitude to all authors, whose research results have been published in our journal, as well as our referees for their in-depth evaluations. Our high standards are maintained through a double blind review process.

We hope that this edition of IJACSA inspires and entices you to submit your own contributions in upcoming issues. Thank you for sharing wisdom.

Thank you for Sharing Wisdom!

Kohei Arai
Editor-in-Chief
IJACSA
Volume 16 Issue 8 August 2025
ISSN 2156-5570 (Online)
ISSN 2158-107X (Print)

Editorial Board

Editor-in-Chief

Dr. Kohei Arai - Saga University
Domains of Research: Technology Trends, Computer Vision, Decision Making, Information Retrieval, Networking, Simulation

---

Associate Editors

Alaa Sheta
Southern Connecticut State University
Domain of Research: Artificial Neural Networks, Computer Vision, Image Processing, Neural Networks, Neuro-Fuzzy Systems

Arun Kulkarni
University of Texas at Tyler
Domain of Research: Machine Vision, Artificial Intelligence, Computer Vision, Data Mining, Image Processing, Machine Learning, Neural Networks, Neuro-Fuzzy Systems

Dr Ronak AL-Haddad
Anglia Ruskin University / Cambridge
Domain of Research : Technology Trends,Communication,Security,Software Engineering and Quality,Computer Networks,Cyber Security,Green Computing,Multimedia Communication,Network Security,Quality of Service

Elena Scutelnicu
"Dunarea de Jos" University of Galati
Domain of Research: e-Learning, e-Learning Tools, Simulation

In Soo Lee
Kyungpook National University
Domain of Research: Intelligent Systems, Artificial Neural Networks, Computational Intelligence, Neural Networks, Perception and Learning

Larbi Boubchir
University of Paris 8
Domain of Research : Artificial Intelligence,Biometrics

Renato De Leone
Università di Camerino
Domain of Research: Mathematical Programming, Large-Scale Parallel Optimization, Transportation problems, Classification problems, Linear and Integer Programming

Xiao-Zhi Gao
University of Eastern Finland
Domain of Research: Artificial Intelligence, Genetic Algorithms

CONTENTS

Paper 1: Development of Web Apps for Users with Special Needs
Authors: Silviya Varbanova, Milena Stefanova, Tihomir Stefanov
PAGE 1 – 10

Paper 2: How Teachers' Gestural Culture Influences Japanese Students' Emotions: A Machine Learning Approach
Authors: Yuka Nishi, Olivia Kennedy, Choi Dongeun, Noriaki Kuwahara
PAGE 11 – 20

Paper 3: Enhancing Approximate Conformance Checking Accuracy with Hierarchical Clustering Model Behaviour Sampling
Authors: Yilin Lyu
PAGE 21 – 37

Paper 4: Enhancing Cyber Security Through Predictive Analytics: Real-Time Threat Detection and Response
Authors: Muhammad Danish
PAGE 38 – 49

Paper 5: A Hybrid Approach to Automatic Timetabling Using Self-Organizing Maps, Secure Convex Dominating Sets, and Metaheuristics
Authors: Elmo Ranolo, Ken Gorro, Pierre Anthony Gwen Abella, Lawrence Roble, Rue Nicole Santillan, Anthony Ilano, Benjie Ociones, Roel Vasquez, Deofel Balijon, Daniel Ariaso Sr., Rose Ann Campita, Robert Jay Angco
PAGE 50 – 58

Paper 6: Integrating Fine-Tuned GPT with Agent-Based Economic Modeling for Transparent Wage Policy Decisions
Authors: Daniel A. Ariaso Sr., Ken D. Gorro, Deofel Balijon, Meshel Balijon
PAGE 59 – 71

Paper 7: A Privacy-Preserving Gaussian Process Regression Framework Against Membership Inference Attacks Using Random Unitary Transformation
Authors: Md. Rashedul Islam, Jannatul Ferdous Akhi, Takayuki Nakachi
PAGE 72 – 80

Paper 8: Autonomous Driving in Adverse Weather: A Multi-Modal Fusion Framework with Uncertainty-Aware Learning for Robust Obstacle Detection
Authors: Zhengqing Li, Baljit Singh Bhathal Singh
PAGE 81 – 86

Paper 9: EcoRouting: Carbon-Aware Path Optimization in Green Internet Architectures
Authors: Handrizal, Herriyance, Amer Sharif
PAGE 87 – 94

Paper 10: Exploring the Future Research Agenda for Health Applications Adoption: A Systematic Literature Review
Authors: Rahmat Fauzi, Adhistya Erna Permanasari, Silmi Fauziati
PAGE 95 – 106

Paper 11: A New Method for Real-Time Fall Detection Based on MediaPipe Pose Estimation and LSTM
Authors: Puwadol Sirikongtham, Apichaya Nimkoompai
PAGE 107 – 113

Paper 12: Design of Marketing Digital Control System Based on the Integration of Big Data Analysis and Machine Learning
Authors: Qiming Li, Songling Du, Shiyuan Zhang
PAGE 114 – 123

Paper 13: Artificial Intelligence in Optometry: Potential Benefits and Key Challenges: A Narrative Review
Authors: Noura A. Aldossary
PAGE 124 – 131

Paper 14: nodeWSNsec: A Hybrid Metaheuristic Approach for Reliable Security and Node Deployment in Wireless Sensor Networks
Authors: Rahul Mishra, Sudhanshu Kumar Jha, Naresh Kshetri, Bishnu Bhusal, Mir Mehedi Rahman, Md Masud Rana, Aimina Ali Eli, Khaled Aminul Islam, Bishwo Prakash Pokharel
PAGE 132 – 142

Paper 15: Exploring Trust Management in Fog Computing: A Comprehensive Review and Future Challenges in Task Offloading
Authors: Liu Feng, Suhaidi Hassan, Mohammed Alsamman
PAGE 143 – 156

Paper 16: AraSpam: A Multitask Deep Neural Network for Spam Detection in Arabic Twitter
Authors: Lulua Alhamdan, Ahmed Alsanad, Nora Al-Twairesh
PAGE 157 – 170

Paper 17: Hierarchical Transformer Residual Model for Pneumonia Detection and Lesion Mapping
Authors: Anupama Prasanth
PAGE 171 – 183

Paper 18: Developing ReAdaBalancer for Load Balancing Optimization in Networked Cloud Computing
Authors: M Diarmansyah Batubara, Poltak Sihombing, Syahril Efendi, Suherman
PAGE 184 – 193

Paper 19: An Augmentation-Based System for Diagnosing COVID-19 Using Deep Learning
Authors: Mohamad Shady Alrahhal, Mohammad A. Mezher, Osamah A.M. Ghaleb, Mohammad Al-Hjouj, Raghad Sehly, Samir Bataineh
PAGE 194 – 205

Paper 20: Navigating the Landscape of Automated Information Extraction for Financial Fund Prospectuses: Survey and Challenges
Authors: Yuyao Xu, Mohamad Farhan Mohamad Mohsin
PAGE 206 – 218

Paper 21: Proactive Cancer Prediction Using IoT and Deep Learning Before Symptoms
Authors: Mohamed Amine Meddaoui, Imane Karkaba, Moulay Amzil, Mohammed Erritali
PAGE 219 – 229

Paper 22: Comparative Analysis of Machine Learning and Deep Learning Models for Handwritten Digit Recognition
Authors: Soukaina Chekki, Boutaina Hdioud, Rachid Oulad Haj Thami, Sanaa El Fkihi
PAGE 230 – 239

Paper 23: Residual DDPG Control with Error-Aware Reward Rescaling for Active Suspension Under Unseen Road Conditions
Authors: Zien Zhang, Abdul Hadi Abd Rahman, Noraishikin Zulkarnain
PAGE 240 – 248

Paper 24: Edge-Guided Multi-Scale YOLOv11n: An Advanced Framework for Accurate Ship Detection in Remote Sensing Imagery
Authors: Yan Shibo, Liu Pan, Abudhahir Buhari
PAGE 249 – 256

Paper 25: The ECTLC-Horcrux Protocol for Decentralized Biometric-Based Self-Sovereign Identity with Time-Lapse Encryption
Authors: N. M. Kaziyeva, R. M. Ospanov, N. Issayev, K. Maulenov, Shakhmaran Seilov
PAGE 257 – 264

Paper 26: Integrating Chatbots into E-Learning Platforms: A Systematic Review
Authors: Victor Sevillano-Vega, Juan Chavez-Perez, Carmen Torres-Ceclén, Orlando Iparraguirre-Villanueva
PAGE 265 – 276

Paper 27: A Hybrid Approach Combining Deep CNN Features with Classical Machine Learning for Diabetic Retinopathy Diagnosis
Authors: Amandeep Kaur, Simranjit Singh, Hardeep Singh, Sarveshwar Bharti, Jai Sharma, Himanshi Sharma
PAGE 277 – 285

Paper 28: Forecasting Currency Exchange Direction with an Advanced Immune-Inspired Model
Authors: EL BADAOUI Mohamed, RAOUYANE Brahim, EL MOUMEN Samira, BELLAFKIH Mostafa
PAGE 286 – 293

Paper 29: Adaptive Ensemble Models for Robust Intrusion Detection in Cloud Environment on Imbalanced Dataset
Authors: Swarnalatha K, Nirmalajyothi Narisetty, Gangadhara Rao Kancherla, Neelima Guntupalli, Simhadri Mallikarjuna Rao, Archana Kalidindi
PAGE 294 – 303

Paper 30: Detection of Leaf Fall Disease in Sembawa Rubber Plantation Through Feature Extraction Model and Clustering Methods
Authors: Alhadi Bustamam, Devvi Sarwinda, Retno Lestari, Ahmad Ihsan Farhani, Harum Ananda Setyawan, Masita Dwi Mandini Manessa, Tri Rappani Febbiyanti, Minami Matsui
PAGE 304 – 312

Paper 31: Tamil Handwritten Character Recognition: A Comprehensive Review of Recent Innovations and Progress
Authors: Manoj K, Iyapparaja M
PAGE 313 – 321

Paper 32: Boosting Deepfake Detection Accuracy with Unsharp Masking and EfficientNet Models
Authors: Radwa Khaled, Hossam M. Moftah, Fahad Kamal Alsheref, Adel Saad Assiri, Kamel Hussein Rahouma, Mohammed Kayed
PAGE 322 – 331

Paper 33: Analysis of the Possibilities of Using LLM Chatbots for Solving Course and Exam Tasks
Authors: Svetlana Stefanova, Yordan Kalmukov
PAGE 332 – 336

Paper 34: Segment-Based Vehicular Congestion Detection Methods Using Vehicle ID and Loss of Expected Time of Arrival
Authors: Mustapha Abubakar Ahmed, Azizul Rahman Mohd Shariff
PAGE 337 – 345

Paper 35: Stock Market Prediction of the Saudi Telecommunication Sector Using Univariate Deep Learning Models
Authors: Hadi S. AlQahtani, Mohammed J. Alhaddad, Mutasem Jarrah
PAGE 346 – 353

Paper 36: A Systematic Review of Multilingual Plagiarism Detection: Approaches and Research Challenges
Authors: Chaimaa BOUAINE, Faouzia BENABBOU, Zineb Ellaky, Amine BOUAINE, Chaimae ZAOUI
PAGE 354 – 372

Paper 37: Towards Explainable and Balanced Federated Learning: A Neural Network Approach for Multi-Client Fraud Detection
Authors: Nurafni Damanik, Chuan-Ming Liu
PAGE 373 – 382

Paper 38: A Novel Multilevel Framework for DoS Detection in SDN
Authors: Rejo Rajan Mathew, Amarsinh Vidhate
PAGE 383 – 391

Paper 39: Mid-Upper Arm Circumference Measurement Using Digital Images: A Top-Down Approach with Panoptic Segmentation Using Mask R-CNN
Authors: Maya Silvi Lydia, Pauzi Ibrahim Nainggolan, Desilia Selvida, Doli Aulia Hamdalah, Dhani Syahputra Bukit, Amalia, Rahmita Wirza Binti O. K. Rahmat
PAGE 392 – 403

Paper 40: Ensemble Learning for Multi-Class Android Malware Detection: A Robust Framework for Family Level Classification
Authors: Mana Saleh Al Reshan
PAGE 404 – 411

Paper 41: Analyzing Cyber Attack Detection in IoT Healthcare Environments Using Artificial Intelligence
Authors: Rawan Marzooq Alharbi, Muhammad Asif Khan
PAGE 412 – 423

Paper 42: Performance Analysis of Proposed Scalable Reversible Randomization Algorithm (SRRA) in Privacy Preserving Big Data Analytics
Authors: Mohana Chelvan P, Rajavarman V N, Dahlia Sam
PAGE 424 – 431

Paper 43: Shared API Call Insights for Optimized Malware Detection in Portable Executable Files
Authors: Mehdi Kmiti, Jallal Eddine Moussaoui, Khalid El Gholami, Yassine Maleh
PAGE 432 – 439

Paper 44: Intrusion Detection Using Machine Learning and Deep Learning
Authors: Fatima Jobran ALzaher, Asma AlJarullah
PAGE 440 – 454

Paper 45: Comprehensive Analysis of Machine and Deep Learning Models for Stock Market Prediction
Authors: Hadi S. AlQahtani, Mohammed J. Alhaddad, Mutasem Jarrah
PAGE 455 – 463

Paper 46: Consumer Adoption of Autonomous Vehicles in China: A Bibliometric Review of Intention Drivers and Perceptions
Authors: Yunluo Zou, Syuhaily Osman, Sharifah Azizah Haron
PAGE 464 – 472

Paper 47: RA-ACS_net Network: A Quantum Optical Reconstruction Method for Ultra-high Resolution Bioimaging
Authors: Lin SHANG
PAGE 473 – 482

Paper 48: Privacy-Preserving Content-Based Medical Image Retrieval Using Integrated CNN Fusion and Quantization Optimization
Authors: Mohamed Jafar sadik, Muhammed E Abd Alkhalec Tharwat, Noor Azah Samsudin, Ezak Fadzrin Bin Ahmad
PAGE 483 – 497

Paper 49: Innovative Model of Tourism on Educational Engineering: Transformation Learning from Experiential to Interactive
Authors: Yurao Yan, Tara Ahmed Mohammed, Hailan Liang, Mingxi Guan
PAGE 498 – 508

Paper 50: DeepIndel: A ResNet-Based Method for Accurate Insertion and Deletion Detection from Long-Read Sequencing
Authors: Md. Shadmim Hasan Sifat, Khandokar Md. Rahat Hossain
PAGE 509 – 515

Paper 51: A Modeling Approach for Strategic Fleet Sizing Under Maritime Sovereignty: Application to the Moroccan National Fleet
Authors: Mohamed Anas KHALFI, Aziz AIT BASSOU, Mustapha HLYAL, Jamila EL ALAMI
PAGE 516 – 527

Paper 52: Prediction of Mining-Induced Subsidence in Saudi Arabia Phosphate Mines Using ANN Method
Authors: Atef GHARBI, Mohamed AYARI, Yamen El Touati, Zeineb Klai, Mahmoud Salaheldin Elsayed, Elsaid Md. Abdelrahim
PAGE 528 – 534

Paper 53: A Review on Image-Based Methods for Plant Disease Identification in Diverse Data Conditions
Authors: Feilong Tang, Rosalyn R Porle, Hoe Tung Yew, Farrah Wong
PAGE 535 – 545

Paper 54: Predictive Modeling for Metro Performance Using MetroPT3 Dataset
Authors: Akshitha Mary A C, R Rakshinee, Stefani Jeyaseelan, Sakthivel V, Prakash P
PAGE 546 – 554

Paper 55: Artificial Intelligence in Diagnostic and Therapeutic Interventions: A Systematic Review of Randomized Controlled Trials
Authors: Oscar Jimenez-Flores, Sandra Pajares-Centeno, Oscar Mejia-Sanchez, Rodrigo Flores-Palacios
PAGE 555 – 562

Paper 56: Transformer-Enabled Smartphone System for Intelligent Physical Activity Monitoring
Authors: Leping Zhang, Fengjiao Jiang, Guopeng Jia, Yue Wang
PAGE 563 – 573

Paper 57: Game Theory-Optimized Attention-Based Temporal Graph Convolutional Network for Spatiotemporal Forecasting of Sea Level Rise
Authors: T M Swathy, K.Ruth Isabels, A. Sindhiya Rebecca, Venubabu Rachapudi, Yousef A.Baker El-Ebiary, Shobana Gorintla, Elangovan Muniyandy
PAGE 574 – 585

Paper 58: Securing Image Messages Using Secure Hash Algorithm 3, Chaos Scheme, and DNA Encoding
Authors: Amer Sharif, Dian Rachmawati, Wilbert
PAGE 586 – 594

Paper 59: Architecting a Privacy-Focused Bitcoin Framework Through a Hybrid Wallet System Integrating Multiple Privacy Techniques
Authors: Lamiaa Said, Hatem Mohamed, Diaa Salama, Nesma Mahmoud
PAGE 595 – 603

Paper 60: VGG-19 and Vision Transformer Enabled Shelf-Life Prediction Model for Intelligent Monitoring and Minimization of Food Waste in Culinary Inventories
Authors: Bindhya Thomas, Priyanka Surendran
PAGE 604 – 615

Paper 61: Analysis of an RGB-D Simultaneous Localization and Mapping Algorithm for Unmanned Aerial Vehicle
Authors: Muhammad Zamir Fathi Mohammad Effendi, Norhidayah Mohamad Yatim, Zarina Mohd Noh, Nur Aqilah Othman
PAGE 616 – 623

Paper 62: In-Depth Comparison of Supervised Classification Models - Performance and Adaptability to Practical Requirements
Authors: Mouataz IDRISSI KHALDI, Allae ERRAISSI, Mustapha HAIN, Mouad BANANE
PAGE 624 – 635

Paper 63: Mobile Applications that Incorporate AI for Information Search and Recommendation: A Systematic Literature Review
Authors: Mijael R. Aliaga, Jhosep S. Llacctahuaman, Carla N. Esquivel, Nemias Saboya
PAGE 636 – 647

Paper 64: Smart Mobile Apps for Responsible Child Management: A Systematic Literature Review
Authors: Daniel Celestino, Harol Medina, Cristian Lara, Nemias Saboya
PAGE 648 – 657

Paper 65: DMME-Driven Product Quality Prediction for Semiconductor Manufacturing
Authors: Alif Ulfa Afifah, Angga Prastiyan, Fahmi Arif, Fadillah Ramadhan
PAGE 658 – 665

Paper 66: AIoT-Based Waste Classification for Solid Waste Management to Accomplish the SDGs
Authors: T. M. Shien, M. Batumalay, Balasubramaniam Muniandy, Pavan Kumar Pagadala, Vinoth Kumar. P
PAGE 666 – 672

Paper 67: A Scalable Machine Learning Framework for Predictive Analytics and Employee Performance Enhancement in Large Enterprises
Authors: Jyoti Singh Kanwar, Ranju S Kartha, Chamandeep Kaur, Behara Venkata Nandakishore, Elangovan Muniyandy, Vuda Sreenivasa Rao, Yousef A.Baker El-Ebiary
PAGE 673 – 684

Paper 68: Deep Learning-Driven Scalable and High-Precision Malaria Detection from Microscopic Blood Smear Images
Authors: N. Kannaiya Raja, Divya Rohatgi, Venkata Lalitha Narla, Ganesh Kumar Anbazhagan, R. Aroul Canessane, Drakshayani Sriramsetti, Yousef A.Baker El-Ebiary
PAGE 685 – 697

Paper 69: Explainable Multimodal Sentiment Analysis Using Hierarchical Attention-Based Adaptive Transformer Models
Authors: Anna Shalini, B. Manikyala Rao, Ranjitha. P. K, Guru Basava Aradhya S, S. Farhad, Elangovan Muniyandy, Yousef A. Baker El-Ebiary
PAGE 698 – 708

Paper 70: A Data-Driven Approach to Achieve Low-Carbon Building Energy Optimization by Using BIM Technology
Authors: Xin Yu, Guoliang Ren, Jie Niu
PAGE 709 – 720

Paper 71: JellyNovaNet-JSO: A Hybrid TabNet–BiLSTM Model for IoT-Based Crop Yield Prediction
Authors: Huang Zhicheng, Zhang Yinjun
PAGE 721 – 735

Paper 72: Functional vs Ethical Drivers in Generative AI Adoption: A PLS-SEM Study in Business Education
Authors: Jorge Serrano-Malebrán, Cristian Vidal-Silva, Paola von-Bichoffshausen, Romina Gómez-López, Franco Campos-Núñez
PAGE 736 – 743

Paper 73: An AI-Driven Approach for Real-Time Noise Level Monitoring and Analysis
Authors: Yellamma Pachipala, L K SureshKumar, Veeranki Venkata Rama Maheswara Rao, Vijaya Chandra Jadala, T.Srinivasarao, D. Srinivasa Rao
PAGE 744 – 751

Paper 74: Deep Learning Meets Bibliometrics: A Survey of Transfer Learning Techniques for Breast Cancer Detection
Authors: Amna Wajid, Natasha Nigar, Hafiz Muhammad Faisal, Olukayode Oki, Jose Lukose
PAGE 752 – 763

Paper 75: An Adaptive Levy Flight Chicken Swarm Optimization with Differential Evolution for Function Optimization Problem
Authors: Wen-Jun Liu, Azlan Mohd Zain, Mohamad Shukor Bin Talib, Sheng-Jun Ma
PAGE 764 – 783

Paper 76: Public Opinion Stage Segmentation in Disaster Events: A Study Based on Multimodal Sentiment Prediction Model
Authors: Xiaogang Yuan, Jiaxi Chen, Dezhi An, Xiang Gong
PAGE 784 – 796

Paper 77: Machine Learning-Based Climate Prediction in Indonesia: A Baseline Experiment
Authors: Faisal Rahutomo, Bambang Harjito
PAGE 797 – 810

Paper 78: Graph-Based Clustering of Short Texts Using Word Embedding Similarity
Authors: Supakpong Jinarat, Ratchakoon Pruengkarn
PAGE 811 – 819

Paper 79: A Prompt-Driven Framework for Reflective Evaluation of Course Alignment Using Large Language Models
Authors: Mashael M. Alsulami
PAGE 820 – 826

Paper 80: AI and 5G Integration for Smart City Energy Systems: A Systematic Literature Review
Authors: TALBI Chaymae, Rahmouni M'hamed, OUAHBI Younesse, ZITI Soumia
PAGE 827 – 834

Paper 81: State-of-the-Art in Software Security Visualization: A Systematic Review
Authors: Ishara Devendra, Chaman Wijesiriwardana, Prasad Wimalaratne
PAGE 835 – 843

Paper 82: Human Versus AI: A Comparative Study of Zero-Shot LLMs and Transformer Models Against Human Annotations for Arabic Sentiment Analysis
Authors: Dimah Alahmadi
PAGE 844 – 851

Paper 83: Simulysis: A Method to Change Impact Analysis in Simulink Projects Based on WAVE-CIA
Authors: Hoang-Viet Tran, Ta Van Thang, Cao Xuan Son, Do Trong Thu, Pham Ngoc Hung
PAGE 852 – 869

Paper 84: Feature Selection and Classification of Microarray Datasets Based on an Improved Binary Harris Hawks Optimization Algorithm
Authors: Guoxia LI, Wen SHI, Jingyu ZHANG, Zhixia GU, Jixiang XU, Yueyue LI, Yaxing SUN
PAGE 870 – 885

Paper 85: Robust Ulcerative Colitis Detection via Integrated Convolutional Feature Encoding, Bidirectional Temporal Context, and Data Augmentation for Class Imbalance
Authors: Dharmendra Gupta, Jayesh Gangrade, Yadvendra Pratap Singh, Shweta Gangrade
PAGE 886 – 893

Paper 86: DALG: A Dual Attention-Based LSTM-GRU Model for Exchange Rate Volatility Forecasting in China's Forex Sector
Authors: Shamaila Butt, Mohammad Abrar, Muhammad Ali Chohan, Muhammad Farrukh Shahzad
PAGE 894 – 901

Paper 87: Parameter-Free Negative Extreme Anomalous Undersampling Techniques on Class Imbalance Problems
Authors: Benjawan Jantamat, Krung Sinapiromsaran
PAGE 902 – 918

Paper 88: Blockchain Enabled Healthcare Supply Chain: Review, Case Study and Future Opportunities
Authors: Muhammad Saad, Kamran Ali, Muhammad Awais Javed, Ahmad Naseem Alvi, Ahmed Alfakeeh
PAGE 919 – 932

Paper 89: An Approach Based on Named Entity Recognition and Semantic Analysis for Recruitment Efficiency and Optimization
Authors: Ismail Ifakir, Noureddine Mohtaram, El Habib Nfaoui, Abderrahim Zannou, Mohammed El Hassouni
PAGE 933 – 941

Paper 90: Towards a Robust DNA Storage System with a Multilayer Approach
Authors: Ayoub Sghir, Manar Sais, Douha Bourached, Jaafar Abouchabaka, Najat Rafalia
PAGE 942 – 949

Paper 91: Hybrid Recommender System for Precision Chemical Application in Banana Cultivation Using Matrix Factorization and Content-Based Filtering
Authors: Ravi Kumar Tirandasu, Prasanth Yalla, Pachipala Yellamma
PAGE 950 – 960

Paper 92: Autonomous Self-Adaptation in the Cloud: ML-Heal's Framework for Proactive Fault Detection and Recovery
Authors: Qais Al-Na'amneh, Mahmoud Aljawarneh, Rahaf Hazaymih, Ayoub Alsarhan, Khalid Hamad Alnafisah, Nayef H. Alshammari, Sami Aziz Alshammari
PAGE 961 – 971

Paper 93: Exploring the Factors Influencing School Dropout: A Logit Model Analysis
Authors: Noaman LAKCHOUCH, Lamarti Sefian MOHAMMED, Mustapha KHALFOUNI
PAGE 972 – 977

Paper 94: Securing the Healthcare Supply Chain Using Blockchain-Enabled Smart Contracts
Authors: Muhammad Saad, Hamail Ashraf, Muhammad Saffi Ullah Khan, Muhammad Awais Javed, Ahmad Naseem Alvi, Ahmed Alfakeeh
PAGE 978 – 987

Paper 95: A Scalable and Privacy-Preserving Hybrid Blockchain Architecture for Secure Healthcare Data Management
Authors: Sanjida Sharmin, Mohammad Shamsul Arefin, Pranab Kumar Dhar, Zinnia Sultana, Sultana Akter
PAGE 988 – 997

Paper 96: Advanced Strategies for Big Data Resource and Storage Optimization: An AI Perspective
Authors: Ayoub Sghir, Ayoub Allali, Najat Rafalia, Jaafar Abouchabaka
PAGE 998 – 1005

Paper 97: YOLOv8s-Swin: Enhanced Tomato Ripeness Detection for Smart Agriculture
Authors: Jalal Uddin Md Akbar, Syafiq Fauzi Kamarulzaman
PAGE 1006 – 1014

Paper 98: Next-Generation Network Security: An Analysis of Threats, Challenges and Emerging Intelligent Defenses Within SDN and NFV Architectures
Authors: Amina SAHBI, Faouzi JAIDI, Adel BOUHOULA
PAGE 1015 – 1024

Paper 99: Bridging Tradition and Technology: Leveraging ERP Systems for Streamlined Supply Chains and Modernized Keropok Lekor Production Management
Authors: Faizah Aplop, Wan Muhammad Ikhwan Wan Mohammad, Muhammad Nasyrul Adly Mohd Afendy, Mustafa Man, Fakhrul Adli Mohd Zaki, Rosaida Rosly, Ismail Abu Bakar
PAGE 1025 – 1030

Paper 100: Enhanced Phishing Website Detection Using Optimized Ensemble Stacking Models
Authors: Zainab Alamri, Abeer Alhuzali, Bassma Alsulami, Daniyal Alghazzawi
PAGE 1031 – 1046

Paper 101: Grey Clustering Algorithm for Urban Air Quality Classification: A Case Study in Lima, Peru
Authors: Alexi Delgado, Katherine Paredes Guerrero, Anderson Carrillo
PAGE 1047 – 1052

Paper 102: Simulation-Driven Improvement of King Khalid University Non-Monthly Entitlement Workflows in AnyLogic
Authors: Elaf Ali Alsisi, Osman A. Nasr, Badriah Mousa Alqahtani, Rodoon Shawan Alnajei
PAGE 1053 – 1058

Paper 103: From Review to Refinement: An Expert-Informed Environmental Diagnostic Model for Stingless Bee Colony Monitoring
Authors: Yang Lejing, Rozi Nor Haizan Nor, Yusmadi Yah Jusoh, Nur Ilyana Ismarau Tajuddin, Khairi Azhar Aziz
PAGE 1059 – 1066

Paper 104: A Secure Authentication Protocol for IoT Devices
Authors: Mohamed Ech-Chebaby, Hicham Zougagh, Hamid Garmani, Zouhair Elhadari
PAGE 1067 – 1076

Paper 105: QoS-Aware Deployment and Synchronization of Digital Twins Over Federated Cloud Platforms for Smart Infrastructure Monitoring
Authors: M V Narayana, Naveen Reddy N, Madhu S, Madhu T, Niladri Sekhar Dey, Sanjeev Shrivastava
PAGE 1077 – 1090

Development of Web Apps for Users with Special Needs

Silviya Varbanova, Milena Stefanova, Tihomir Stefanov
Faculty of Mathematics and Informatics, St. Cyril and St. Methodius University, Veliko Tarnovo, Bulgaria

***Abstract*—This study presents software solutions and prototypes of converters designed to assist individuals who suffer from visual impairments or deafness. The prototypes were developed in a laboratory environment using modern programming technologies, with their applicability focused on contexts such as education, employment, and online shopping. The converter prototypes are designed to transform textual information into Braille or sign language, depending on users' needs. Other solutions facilitate auditory interpretation when working with assessment materials, thereby helping the sightless to access information more easily. The research methodology combines synthesis and analysis of existing information from related research works. The choice of programming technologies was carefully considered to ensure the implementation of more accessible functionalities in the developed applications. In the teaching process, the authors of the study motivate their students to develop programming and analytical skills. The results achieved are based on student project work in the design and implementation of software prototypes to assist users with hearing or visual impairments. The prototypes created are aligned with scenarios for use in education and everyday activities, which expand the practical relevance of the study. Importantly, the presented applications also benefit people without disabilities by promoting communication that is more effective and understanding in cases of visually or hearing impaired.**

***Keywords*—Braille; sign language; deafness; visually impaired; software prototypes**

I. INTRODUCTION

The mobility, communication, education, and employment of those with hearing or visual impairments are more challenging than those with no impairments. In the past, support for these activities was provided mainly through personal assistants or mechanical aids [1]. Today, however, electronic applications and computer software have significantly enhanced the independence of these users. Modern digital devices have made daily activities – such as navigation, shopping, and communication – progressively more accessible for people with different disabilities [2].

Users with hearing or visual impairments seek to minimize the limitations they encounter in everyday activities such as learning, shopping, and working. To improve existing solutions, many actively collaborate with scientists and developers as part of dedicated teams. In numerous cases, these users themselves propose innovative approaches and more accessible functionalities, directly contributing to the development of software products that better address their needs. Based on the research into the needs of specific user groups, they are now increasingly encouraged to use mobile and computer devices. The convenience that modern specialized applications developed for them offer is also significant. This significantly reduces the difficulties in taking in and passing on information when performing various activities [3].

One of the current challenges being addressed is finding effective solutions to increase the participation of students and workers with special needs. In practice, not all individuals with disabilities can attend specialized educational institutions and subsequently find professional fulfillment. A significant proportion instead study in mainstream educational institutions, where learning is made more accessible through the use of innovative modern technologies. Advancements in digital technologies are occurring at an unprecedented pace and have had a profound impact on education and professional development, including for people with special needs [4]. These technologies empower anyone who seeks to acquire knowledge and succeed in their chosen field.

For example, in the past, visually impaired individuals typically relied on Braille documents and specialized teaching aids for their education. Today, however, modern computers and mobile devices enable them to independently read and write texts in standard formats, beyond just Braille [5]. As a result, users gain greater confidence in accessing and understanding textual, graphical, and semantic information. This advancement is largely made possible by the availability of screen readers, speech synthesizers, and various conversion tools [6]. The development of assistive technology for those with visual impairment is predicted to continue, according to researchers [7]. The authors assert that this growth could have a substantial impact on people's lives in a way that was previously impossible. Since the final decade of the twentieth century, the development and dissemination of such software have played a crucial role in supporting visually impaired users. These software converters transform textual information displayed on a device screen into audio output, allowing blind or visually impaired individuals to listen to and comprehend written content, as well as to review and edit the text they produce themselves.

Software applications are being developed to support individuals with hearing impairments in their education and professional growth. Although the number of such applications remains limited relatively, notable progress has been made in recent years. Software converters can transform gesture-based information displayed on computer or mobile screens into textual content and vice versa [8].

Additionally, to facilitate communication between hearing-impaired individuals and those without hearing impairments, systems are being designed to recognize and translate sign language into audio output and, conversely, convert spoken language into sign language representations. These advancements contribute significantly to improving social interactions, educational opportunities, and professional development for people with hearing impairments [9].

Scientists are working towards the development and implementation of speech and sound visualization systems for people with hearing impairments. Detecting signal events is the main tool for improving the semantic perception of given information. For example, by using other senses to recognize information, such as visual, acoustic, and tactile [10].

Another significant issue for the hearing impaired is the correct perception of musical sounds so that they are not limited when learning or working with music, video or audio data [11]. Scientists are researching, analyzing and searching for more accessible techniques for this purpose. The problem is similar when working or studying with mathematical terminology in areas such as mathematics, physics, and economics [12]. Here, the correct use of sign language by learners and trainers is important. When used incorrectly, semantic errors inevitably occur. For this purpose, more and more analogues are being sought today for better techniques for practicing sign language in communication. The age limit of hearing-impaired users is also important here, which turns out to be a serious barrier.

Today, regional libraries, bookstores, educational institutions, and office centers are increasingly offering audio-electronic catalogs featuring both specialized literature and fiction. Additionally, multimedia video and audio presentation materials are often supplemented with subtitles, thereby partially improving information accessibility for hearing-impaired users.

The authors of this study are actively contributing to developments in this field. The content of the study is organized as follows: The Introduction section outlines the need for software solutions specifically designed for people with hearing impairments or visual disabilities. The Related Work section provides an overview and analysis of relevant articles connected to assistive technologies and software for users with hearing or visual impairments. The Methodology section describes the analysis and discussions conducted on the topic, along with the results of collaborative work involving students, academics and programmers. The Results section presents various projects developed through the phased implementation and testing of program code, highlighting software solutions created for these users. The Discussion section summarizes the main findings of the study and examines the challenges encountered by the authors. Finally, the Conclusion section offers a concise summary of the study's contributions and insights.

II. Related Work

The perception of information from the surroundings could be longer-lasting for individuals with special needs. Software is primarily designed for common users by developers, and people with hearing or vision impairments find it hard to use. Specialized applications have been developed for these user groups to read, hear, or understand graphical information.

For example, electronic readers for computer devices with built-in multilingual text synthesizers such as NVDA, JAWS, VoiceOver [13], [14]. Similar software applications are designed for the Windows OS and support browsers such as Mozilla Firefox and Google Chrome. Their use is associated with facilitating the work of the visually impaired with office programs, in converting text to braille, and in internet communication. According to available statistics, Fig. 1 presents some of the most commonly used text readers, by usage during the period. For each of the screen readers, advantages and disadvantages are reflected [15].

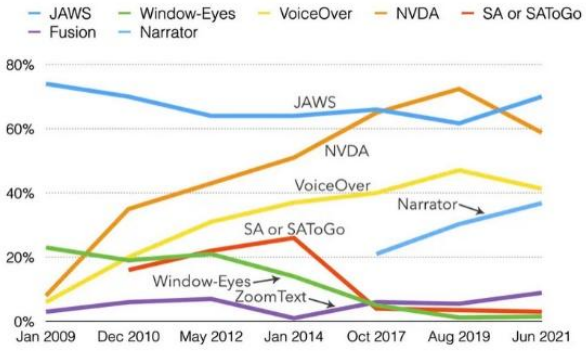


Fig. 1. Screen readers – popularity and usage.

When individuals with visual impairments use text-to-speech synthesizers, some of their built-in features may be difficult to access or not preferred by users. To address this, separate systems are being developed that are simpler to operate and do not require lengthy setup or installation processes. Another emerging trend aimed at helping blind users manage electronic messages more easily is the creation and deployment of systems such as 'talking email' [16], [17].

Currently, there are still relatively few online converters available for transforming text into braille (see Fig. 2) and vice versa [18]. The existing programs, however, are practical and favored by people with low vision who rely on them. One key advantage is that, with access to a Braille printer and suitable paper, the converted text can be printed. This allows visually impaired individuals who are proficient in Braille to independently read printed materials [18], [19].

The reverse process is also achievable through online tools. Braille characters are converted to text, which often has one specific drawback. When the text is converted, the result is displayed mostly in lowercase (Fig. 3, [20]). This problem is solved partially in the prototype described below in Task B1.

Individuals who are blind often encounter difficulties in perceiving typography and color layouts [21]. Although this issue remains an active area of research, efforts continue to identify better solutions. The challenge related to interpreting 2D tactile graphics and 3D models – particularly in fields that involve technical drawings – has been partially addressed through the development of innovative assistive devices. One

such device incorporates a camera and plays pre-recorded audio descriptions when the user touches specific parts of a graphic or layout with their fingers. However, despite its potential benefits, this device is currently not widely accessible for personal use due to its high market cost [22].

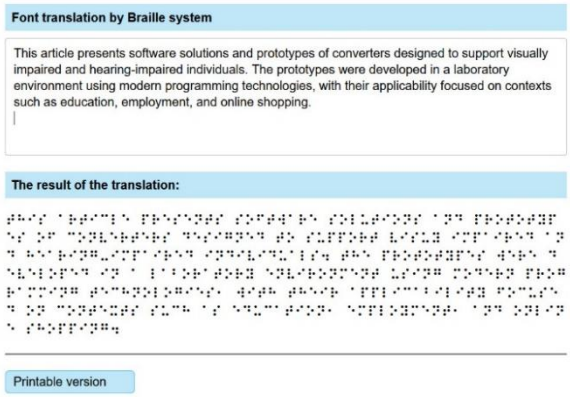


Fig. 2. Transforming text to Braille characters.



Fig. 3. Transforming Braille characters to text.

A significant challenge remains in the rapid conversion of auditory signals into forms accessible to people with hearing impairments. This issue is not limited to educational institutions, office buildings, or retail spaces, but also extends to vehicles. When there is a delay in responding to warning signals, the risk of accidents increases. The authors, who describe the creation of an application by using Tensor FlowLite, which reduces energy consumption and improves response time [23], present a specific advantage in this.

The focus is on exploring papers about assistive technologies supporting people with hearing or visual impairments. The application of these technologies in the education of these user groups is another important criterion in the selection process of the referenced related works.

The authors' interests in this area are influenced by the nature of their teaching activities. They aim to enhance the skills of their students in building web-based applications, while also developing their critical self-assessment and empathy towards the specific users for whom these applications are meant.

Table I provides a comparison of the study with similar ones and also highlight specific contributions related to educational activities.

TABLE I. COMPARISON OF THE PRESENTED CONTRIBUTIONS AND APPLICABILITY IN EDUCATIONAL ACTIVITIES

Survey	Contributions Described	
	In the Results Section	***Regarding Educational Activities***
Zhu et al., 2025 [4]	The important roles of accessible technologies in 'Blind Colleges', yet also reveal multiple practical challenges due to inaccessibility are highlighted.	The main contribution for HCI communities is raising awareness of the imbalance in educational technology by promoting the importance of their innovation.
Apu et al., 2021 [5]	The offered Refreshable Braille Translator converts text from various languages into Braille characters, enabling those with visual impairments to learn and read in digitally.	Yes, significant benefits in the field of education are presented.
Murthy et al., 2022 [6]	Various tools used for translating Braille documents to English text are evaluated by the authors, applying their work to single-sided Braille documents.	Applicable to individuals with visual impairments who use only Braille documents as a medium for education.
Da Silva Dessbesel et al., 2025 [12]	The findings indicate that teachers' understanding of sign language can have a significant impact on the teaching of mathematics in deaf education.	Yes. The study is examining the role of sign language in teaching and learning mathematics for deaf students.
Nethravathi et al., 2022 [23]	The present application system aims to identify potential daily objects in human surroundings and provide speech feedback to users about the objects around them.	Not specified. Suitable for everyday activities. The created application uses Tensor FlowLite, which enhances response time and requires low power, making it suitable for transferable apps.
Proposed	Several translators and converters are presented to transform information into facial gestures, Braille, audio, and vice versa, personalized to the users' specific needs, aiming to facilitate data perception and communicate.	Mostly prototypes that are developed during student training are described.

Based on research and discussions with visually impaired and hearing-impaired users, the authors of this study propose solutions for developing more accessible applications. Such projects can be implemented both within software companies and in educational institutions and research laboratories, to benefit people with visual or hearing impairments. In this context, practical solutions and projects are currently being developed within computer-oriented disciplines at the Faculty of Mathematics and Informatics at the University of Veliko Tarnovo. These projects primarily focus on creating embedded voice readers, converters that transform text into mimic or Braille information, and tools for facilitating the perception of images and multimedia content.

III. Methodology

The research involves studying various publications, as well as various software and applications, for the benefit of individuals with visual or hearing impairments. The methodology relies on the synthesis and analysis of existing information, as well as the identification of concepts from practice.

The information in this study is summarized on the basis of the collaborative work of faculty members from the University of Veliko Tarnovo and programmers. Based on the meetings, accessible functionalities in prototype software products were planned. In parallel, software application designs were designed and developed in a laboratory environment. Advanced programming techniques and web technologies were used to design the reviewed prototypes. The prototypes were primarily made for individuals who have hearing or visual impairments.

The authors of the study aim to highlight the need for projects and digital applications that support people with hearing or visual impairments, helping them adapt more effectively in both work and education. Their goal is to encourage young professionals and individuals with these impairments to collaborate on developing practical solutions. By involving not only researchers and developers but also IT educators and students, the authors hope to create effective tools to reduce barriers in communication, learning, and employment for people with visual or hearing challenges.

In the Faculty of Mathematics and Informatics, undergraduate and graduate students develop software projects in a variety of subjects, often forming working teams. They actively participate in developing prototypes of electronic applications and web pages. Through their expertise, they help address and mitigate some of the challenges faced by people with disabilities. These applications were created during seminar classes in core disciplines and tested in real-world environments. For the practical implementation of the coding aspects of the projects, modern technologies were employed, including markup and stylesheet languages for adaptive design, scripting languages, frameworks, databases, and high-level programming languages.

IV. Results

This section presents software projects and prototypes designed to assist users with hearing or vision impairments, which were implemented during the training of students in core programming disciplines. Developed in a laboratory environment, the projects were tested using modern software testing tools. Some of the projects require the development of software for working with scientific and literary texts. The implementation of other prototypes in the computer technology field is aimed at conducting training tests. Others are in the field of online shopping.

The tasks and projects below, which are grouped into three categories, have been implemented and tested in a laboratory environment. For each prototype, the minimum requirements and implementation criteria have been defined beforehand.

A. Projects for the Benefit of Hearing-Impaired Users

As part of their training in core disciplines, student teams also focus on addressing the challenges faced by people with hearing impairments when using software products. Since these users have limited or no ability to hear, solutions that convert speech into text information – or even into facial gestures – can significantly enhance their user experience. Such projects aim to improve accessibility and inclusivity by enabling hearing-impaired users to interact more effectively with digital applications.

Task A1. To implement a web application for converting facial gestures into Cyrillic and vice versa.

Minimum functional requirements:

- Visual interpretation of the text converted into gestures (see Fig. 4), represented by the corresponding images [24].

The team is currently working on projects to implement computer games for users with impaired vision or hearing.

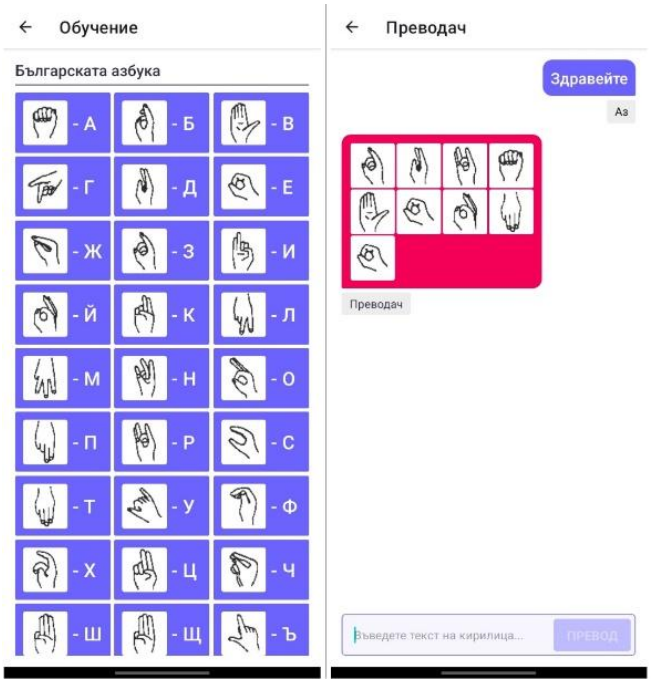


Fig. 4. Screens for visualizing gestures for letters and expressions in Bulgarian.

B. Projects that Meet the Needs of Visually Impaired Users

Task B1. To implement a prototype text-to-speech converter.

After entering a random string, the converter must satisfy the following minimum functional requirements for the final product:

- Count of characters entered, presented by a number;
- Convert text by capitalizing the first letter of each word;
- Splitting the text into paragraphs, which is particularly useful for essays, course projects or theses.

The program implements text recognition from various spoken languages, allowing for an unlimited amount of text to be entered into the field. A web visualization in Fig. 5 shows a specific instance of how the text converter works, with simplified navigation and an input field. Part of the programming code for capitalizing text is presented in Fig. 6.

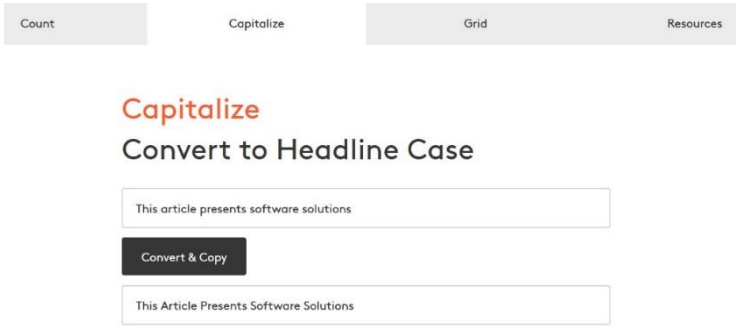


Fig. 5. Web visualization of text converter.

```
<div class="container main">
  <div class="row header">
    <div class="col-sm-8 col-sm-offset-2">
        <h1 class="large">Capitalize</h1>
        <div class="desc">Convert to Headline Case</div>
    </div>
  </div>
  <div class="row">
    <form id="tester" action class="col-sm-8 col-sm-offset-2">
      <input id="untitled" class="input" type="text" placeholder="Paste/type your text here...">
      <!-- <button type="submit1" class="submit" value="Convert">Convert</button> -->
      <button id="copyButton" type="submit2" class="submit">Convert & Copy</button>
      <input id="titled" class="input" type="text">
      <p id="confirmation" class="cap-note">Text has been copied to your clipboard.</p>
    </form>
  </div>
</div>
```

Fig. 6. Text converter – code snippet of realized functionality.

Task B2. To design and develop a text-to-speech converter.

Minimum functional requirements – for randomly entered text, the software solution should allow users to do independently:

- Start hearing the text readings aloud by selecting a button;
- Stop the playback/listening of the text by pressing a button.

Additional convenience of the program implementation offers users the opportunity to work in other windows on other tasks, while at the same time they can listen to the voice reader. The prototype being developed does not need a web version. The JavaScript fragment code for an example implementation of this task can be seen in Fig. 7.

Task B3. To implement a sample test in the discipline Web Technologies with audio recognition.

Functional requirements:

- Convenient interface – presence of buttons, designed for the user to independently move to the next page or question;
- Voice playback of the question, as well as the possible answers;
- Calculation and display of the final result – after filling in the answers.

The test's programmatic implementation can vary depending on whether each question with possible answers is presented on a separate page or not.

If the questions are displayed on distinct pages, there are still possibilities for the final prototype. Case 1: when starting the test, from the current page, each question should be played out in voice in advance, as well as the possible answer options. The user is given no limit on the time they have to choose a response. Clicking on a button takes the user to the next page after indicating their answer (see Fig. 8).

By selecting the answer to the last question, the result of the Test is automatically displayed which is realized with program code shown on Fig. 9.

```
/* Stop speaking */
const stopSpeaking = () => {
    synth.cancel();
    stopButton.disabled = true;
    speakButton.disabled = false;    };

/* Provide voice feedback for button hover */
const speakHint = (message) => {
    const hint = new SpeechSynthesisUtterance(message);
    hint.voice = synth.getVoices()[0];
    hint.rate = 1;
    hint.pitch = 1;
    synth.speak(hint);
};

/* Announce current slider value */
const announceRangeValue = (type, value) => {
    const message = `${type} is set to ${value}`;
    const utterance = new SpeechSynthesisUtterance(message);
    utterance.voice = synth.getVoices()[0];
    utterance.rate = 1;
    utterance.pitch = 1;
    synth.speak(utterance);     };

/* Announce limit when sliders reach min or max */
const announceLimit = (type, value, min, max) => {
    if (value == min) {
        speakHint(`${type} is at minimum`);
    } else if (value == max) {
        speakHint(`${type} is at maximum`);
    }      };
```

Fig. 7. Text-to-speech converter – JavaScript fragment code.

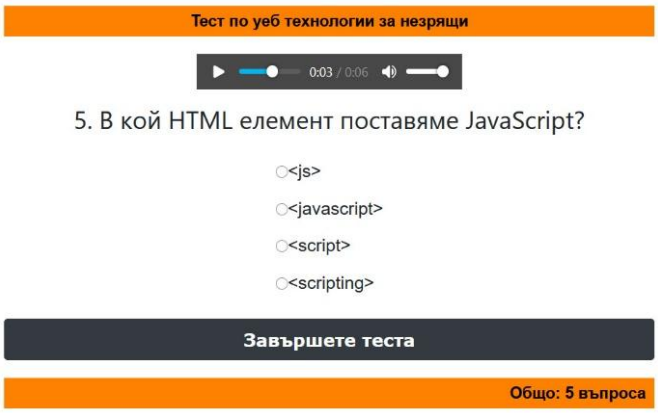


Fig. 8. Preview of the last question in the test for Case 1.

```
<form class="questionForm" id="q5" data-question="5">
    <audio controls>
        <source src="audio/въпрос5.mp3" type="audio/mp3">
    </audio>
    <h3 title="5. В кой HTML елемент поставяме JavaScript?">5. В кой HTML елемент поставяме JavaScript?</h3>
    <ul class="main-ul-list">
        <li class="5-1" title="js"><label><input type="radio" name="q5" value="a" alt="js"/>&lt;js&gt;</label></li>
        <li class="5-2" title="javascript" ><label><input type="radio" name="q5" value="b" alt="javascript" />&lt;javascript&gt;</label></li>
        <li class="5-3" title="script"><label><input type="radio" name="q5" value="c" alt="script"/>&lt;script&gt; </label></li>
        <li class="5-4" title="scripting"><label><input type="radio" name="q5" value="d" alt="scripting"/>&lt;scripting&gt;</label></li>
    </ul>
    <button id="submit" class="btn btn-dark btn-lg btn-block" title="Завършете теста">Завършете теста</button>
        <audio preload="auto" id="5-1"class="d-none">
        <source src="audio/5-1.mp3" controls></source>
    </audio>
    <audio  preload="auto" id="5-2" class="d-none">
        <source src="audio/5-2.mp3" controls></source>
    </audio>
    <audio  preload="auto" id="5-3" class="d-none">
        <source src="audio/5-3.mp3" controls></source>
    </audio>
    <audio preload="auto"  id="5-4" class="d-none">
        <source src="audio/5-4.mp3" controls></source>
    </audio>
</form>
```

Fig. 9. Program code to display test result for Case 1.

Case 2: when starting the test, a voice is heard playing the specific question. Then, when the mouse is moved over each answer, it is spoken accordingly. In this way, the user has the opportunity to indicate an answer that he or she judges to be correct, after careful consideration and repeated listening to the choices. Users move self-directed to the next page. The result of the test is calculated and displayed only after the last question is answered and the 'Finish quiz' button is clicked (see Fig. 10 and Fig. 11).

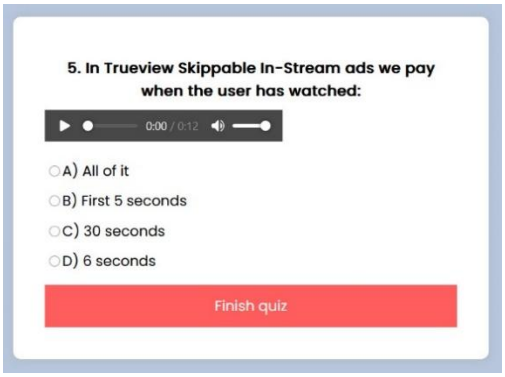


Fig. 10. Preview of the quiz question in Case 2.

```
<form class="questionForm" id="q5" data-question="5">
    <h3>5. In Trueview Skippable In-Stream ads we pay when the user has watched:</h3>
    <audio controls>
        <source src="question5.mp3" type="audio/mp3">
    </audio>
    <ul>
        <li><input type="radio" name="q5" value="a" />A) All of it</li>
        <li><input type="radio" name="q5" value="b" />B) First 5 seconds</li>
        <li><input type="radio" name="q5" value="c" />C) 30 seconds</li>
        <li><input type="radio" name="q5" value="d" />D) 6 seconds</li>
    </ul>
    <button id="submit">Finish quiz</button>
</form>
```

Fig. 11. Fragment of the program code for Case 2.

If the test questions and their possible answers are presented on a single page, the program can guide users through the following process: all questions, along with the answer options, should be read aloud using text-to-speech functionality. Users would then select their answers with voice assistance or other accessible input methods. After completing the selection, users can press a designated button to calculate and display the results. Fig. 12 below shows an example of the code used to check the correctness of the answers, implemented with JavaScript-based technologies.

```
function checkAnswer(q){
    if(q=="q1"){
        var submitted=$('input[name=q1]:checked').val();
        if(submitted==sessionStorage.a1){
            score++;}}
    if(q=="q2"){
        var submitted=$('input[name=q2]:checked').val();
        if(submitted==sessionStorage.a2){
            score++;}}
    if(q=="q3"){
        var submitted=$('input[name=q3]:checked').val();
        if(submitted==sessionStorage.a3){
            score++;}}
    if(q=="q4"){
        var submitted=$('input[name=q4]:checked').val();
        if(submitted==sessionStorage.a4){
            score++;}}
    if(q=="q5"){
        var submitted=$('input[name=q5]:checked').val();
        if(submitted==sessionStorage.a5){
            score++;}
        $('#results').html('<h3>You answered '+score+'/5 questions correctly.</h3>
        <a href="index.html">Reload Quiz</a>');}
    return false;
}
window.addEventListener('load',setAnswers,false);
```

Fig. 12. JavaScript code to check answers in case all questions are on one page.

C. Projects for the Benefit of Users with Impaired Hearing or Vision

Task C1. To implement an adaptive version of an online store for mobile devices.

Functional requirements:

- Easy access to the navigation menu (see Fig. 13);
- Voice playback of a description or functionality for an element (image or text) in the online store – when hovering over this element with the mouse;
- The included multimedia elements should also be adapted for people with hearing impairments.

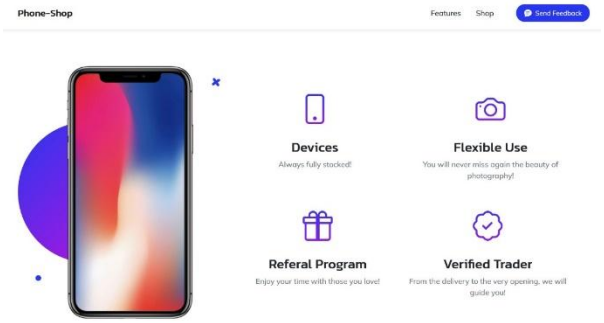


Fig. 13. Online store home screen preview on desktop version.

Below are selected prototypes of student projects, developed in laboratory settings and tested in real-world environments.

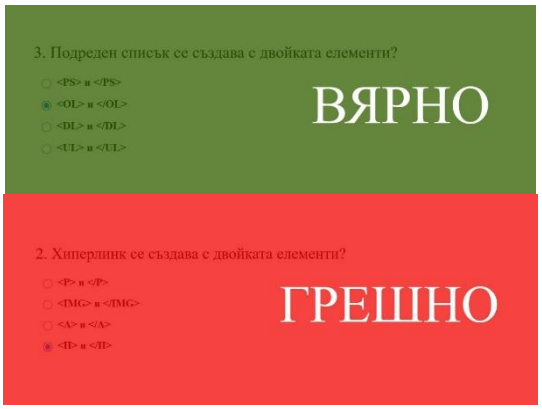
Project one. Specialized test for the level of HTML proficiency, aimed at users with hearing or vision impairment.

A specialized test has been developed for the benefit of people with hearing impairment. Each of the questions is on a separate page and has four answers. After selecting an answer, the participant is automatically taken to the next page.

As a final result of the Test, a report of accuracy is generated, combined with a bright color scheme that hearing-impaired users can see. The result is also presented as a voice report/message with audio layout for correct or incorrect answer – for visually impaired users. The text part of the Test is in Bulgarian, and it is possible to translate it into other languages. The following Fig. 14 shows the result of the indicated answers to two of the questions in the test, respectively – for correct or incorrect answer.

Project two. A Web Design test for the benefit of users with hearing or visual impairments.

A similar test to the ones presented in the study above has been developed, but with a different design. Each of the questions is on a new page and is read with a voice embedded reader. Users have a choice of three possible answers. As a result of taking the Test, a text message or an audio signal is generated indicating truth or falsity. The text part of the Test is in English (see Fig. 15), and the program code fragment is shown in Fig. 16.



(a)

```
<body>
    <h1 class="entry" data-entry="">НАЧАЛО НА ТЕСТ</h1>
    <div class="test-wrapper hidden" data-test="wrapper">
        <p data-test="question"></p>
        <div class="option-answers" data-test="answers">
          <input type="radio" id="answer1" name="answer">
          <label for="answer1"></label><br>
          <input type="radio" id="answer2" name="answer">
          <label for="answer2"></label><br>
          <input type="radio" id="answer3" name="answer">
          <label for="answer3"></label><br>
          <input type="radio" id="answer4" name="answer">
          <label for="answer4"></label>
        </div>
        <input type="hidden" data-test="correct-answer">
    </div>
    <div class="answer-state" data-answer="state">
    </div>
    <span class="answer-state-bg hidden" data-answer="state-bg"></span>
    <div class="final-results-wrap hidden" data-answer="final">
      <span class="final-results"></span>
      <a href="javascript:void(0)" data-test="retake">Опитай пак</a>
    </div>
</body>
```

(b)

Fig. 14. Project one: (a) Answer options; (b) Part of the program code.



Fig. 15. Project two – version of answers.

```
<audio id="startsound">
  <source src="assets/audio/start.mp3" />
</audio>

<script type="text/javascript">
                var callTimeout = function () {
                        setTimeout("location.href = 'test.html';",1500);
                }
                window.onload=function() {
                        const mySound = document.getElementById("startsound");
                        const startButton = document.getElementById("startb");
                        startButton.addEventListener("click", function() {
                                mySound.play();
                        })
                }
</script>
```

Fig. 16. Project two – program code fragment.

We believe that the implementation of the tasks set in this way has helped to improve the knowledge and practical skills of the students both in programming and in more successful communication in their contacts with people with hearing or vision problems.

V. Discussion

The rapid development of technology today is transforming and being applied across all areas of society. Despite this progress, significant limitations remain for people with special needs. There is a pressing need for appropriate technologies and software applications to support visually and hearing-impaired individuals in both education and employment.

For some, the process of perceiving and processing information can be time-consuming and challenging. Moreover, not all individuals have the opportunity to study in specialized institutions or resource centers staffed by experienced teachers and researchers. Securing permanent employment is also often a lengthy and difficult journey. Contributing to these challenges are various barriers, including financial constraints, geographical distance, transportation difficulties, family responsibilities, and other personal or societal factors.

This study describes working solutions for a more accessible perception of text or audio information for people with visual or hearing difficulties. The prototypes are mainly intended for training, work, or communication with individuals without disabilities. These activities are significantly facilitated by the availability of modern computer devices and web applications. Thus, the visually-impaired and the hearing-disadvantaged adapt better to society and communicate more easily. Ordinary users could understand the reverse process of converting Braille or mimicking information into text or audio through the few existing software converters.

Individuals with visual impairments and partially sighted individuals often have trouble understanding multimedia information. When it comes to studying or applying for a job online, for instance. It is common for the websites for these activities to be not fully accessible to text-to-speech software in many cases. The issue is significant, and to reduce it, teams are incorporating web developers who are visually impaired, scientists, or people with hearing impairments.

The results of this study can be summarized in several key areas corresponding to the main stages of the work process. Fig. 17 shows a high-level diagram of the overall study. The primary stages involve designing the tasks and functionalities of the prototypes created by students. As a result of critical analysis and in an effort to overcome identified gaps, ideas for the team's future work are determined.

In future research, the author's team will focus on specialized empirical studies, similar to those presented in [25], [26], with teams of students participating in the development of software applications for the studied group of users. In future developments, the authors plan to focus on improving voice control functionality for users with visual impairments. For the benefit of users with color blindness, the team's efforts will be directed at improving the adaptive design for color layout. This would also limit the problems for users with color blindness when using control buttons and navigating in applications. Another issue that will be addressed in future research is related to functionalities for users with motor delays, aimed at working without using a computer mouse.

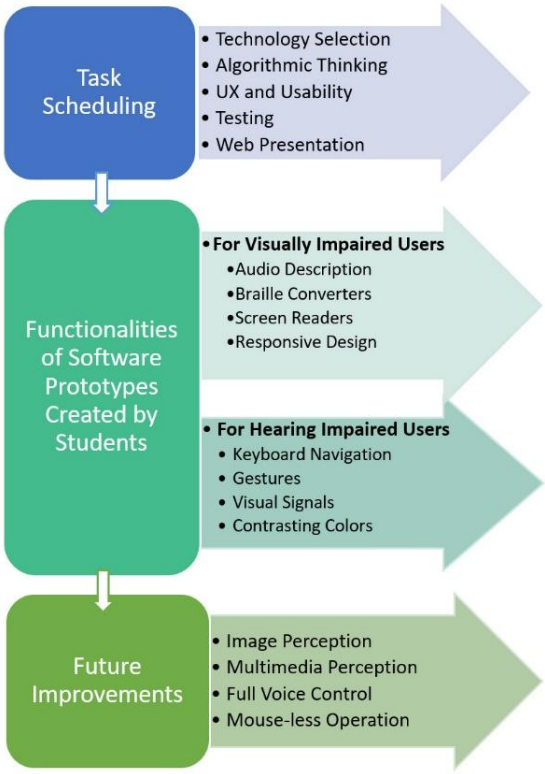


Fig. 17. Flowchart of the research workflow.

An important problem that is being worked on is the inability to accurately perceive images, graphics, or video data by people with 100% blindness or deafness. Currently, there is no standard technology that permits the perception of images, diagrams, or video data through hearing or sight alone. Software applications are still unable to accurately 'describe' the meaning/semantics of a given image or video. The SnapStick authors in this direction are making significant progress [26]. The SnapStick application, approbated by Shafique et al. (2025), enables the description of images of people, interiors, or outdoor environments. The authors evaluate the software using the System Usability Scale (SUS) methodology, focusing on completeness, accuracy, relevance, consistency, and photographs' details. They demonstrate that the application successfully handles object description from a photograph, even recognizing different human emotions from images of people [26]. However, usually in the content of web pages that include images, screen readers or speech synthesizers pronounce them as 'picture', or read out the text set by the developers in the 'alt' / 'title' attributes in the code part. Only with the help of another person – an assistant, can the visual information of a visually impaired person be described. In many cases, an accurate or complete description is impossible.

Another significant obstacle that people with low vision face is when using online shopping sites. For example, when specifying an address and delivery date, the inability to ignore advertising banners or to view images.

Nowadays, work is also being done on the reverse process – to generate an image from a description. For this purpose, AI-based image-to-text generators are used [27].

VI. Conclusions

This study aims to promote current applications and software designed for groups of users with impaired vision or hearing. Working prototypes and solutions are offered that support the work and training of people with disabilities, as well as their ability to better communicate and socialize in society. For people with hearing impairments, auditory information is not fully accessible; they have difficulty perceiving spoken communication. Instead, they rely more on visual information and sign language. Conversely, people with visual impairments face difficulties in accessing visual content. To address this, various translators and converters have been created to transform information into Braille, facial gestures, audio, and vice versa – tailored to the user's specific needs. This facilitates communication and data perception. Nowadays, every user aims to get the required information with as few mouse clicks as possible. This is especially relevant for users with special needs. For them, it would be beneficial to reduce clicks to a minimum, as is the case for people without disabilities. The software prototypes presented are notable for this key feature, and this is one of the contributions to the development. Screen readers and Braille to text converters are being developed for visually impaired users to make them easier to use via computer and mobile devices.

To improve quality of life and facilitate social inclusion, scientists from various fields continue to explore better solutions for people with hearing or visual impairments. We will continue the studies in this area in future academic activities with our students. Based on thorough analysis, research, and identified needs, new functionalities and software solutions are being developed to create more accessible products for diverse users. By understanding the challenges faced by these users, programmers, researchers, and educators continuously refine and enhance these specialized products. The prototypes incorporate features such as speech output and text descriptions, making software interfaces easier to navigate. The focus of the study is on designing and implementing assistive software prototypes, particularly through student-focused projects. Various scenarios for application in education, employment, and everyday activities are outlined, which expand the practical significance of the research.

A widespread application of Artificial Intelligence and the Internet of Things is anticipated in the design and development of electronic devices and applications in the near future. This trend is expected to drive the creation and enhancement of increasingly sophisticated solutions that benefit users with hearing or visual impairments.

References

[1] T. Stefanov, S. Varbanova, M. Stefanova, "An Overview of some popular devices and technologies designed for blind and visually impaired people," *2022 International Conference Automatics and Informatics (ICAI)*, pp. 173–178, Oct. 2022, doi: 10.1109/icai55857.2022.9959981.

[2] W. Pacardo, "A Modified Cane for the Blind using Computer Vision," *Journal of Innovative Technology Convergence*., vol. 6, no. 3, pp. 51–66, Sep. 2024, doi: 10.69478/jitc2024v6n3a06.

[3] T. Stefanov, S. Varbanova, M. Stefanova, "Mobile Devices Supporting People with Special Needs," *International Journal of Advanced Computer Science and Applications (IJACSA), 13*(11), 2022. doi: 10.14569/IJACSA.2022.0131122.

[4] X. T. Zhu, Z. Qiu, Y. Wei, J. Wang, and Y. Jiao, "Understanding the Practice, Perception, and Challenge of Blind or Low Vision Students Learning through Accessible Technologies in Non-Inclusive 'Blind Colleges,'" *arXiv.org*, Jan. 13, 2025. https://arxiv.org/abs/2501.07736v1.

[5] F. S. Apu, F. I. Joyti, M. A. U. Anik, M. W. U. Zobayer, A. K. Dey, and S. Sakhawat, "Text and voice to Braille translator for blind people," *2021 International Conference on Automation, Control and Mechatronics for Industry 4.0 (ACMI)*, pp. 1–6, Jul. 2021, doi: 10.1109/acmi53878.2021.9528283.

[6] V.V. Murthy, M. Hanumanthappa, "Comparative study of various Tools used in Braille to Text Translation," *International Journal of Science & Engineering Development Research* (www.ijrti.org), ISSN: 2455-2631, 2022, *Vol.7, Issue 7*, pp. 1479–1484, July-2022, Available: http://www.ijrti.org/papers/IJRTI2207259.pdf.

[7] A. Bhowmick and S. M. Hazarika, "An insight into assistive technology for the visually impaired and blind people: state-of-the-art and future trends," *Journal on Multimodal User Interfaces*, vol. 11, no. 2, pp. 149–172, Jan. 2017, doi: 10.1007/s12193-016-0235-6.

[8] Hand Talk, "Meet the Hand Talk Sign Language Translator App - Hand Talk," *Hand Talk - Learn ASL today*, Feb. 22, 2024. https://www.handtalk.me/en/blog/meet-the-hand-talk-sign-language-translator-app/.

[9] B. Alsharif, M. Ilyas, "Internet of Things Technologies in Healthcare for People with Hearing Impairments," In: Wang, S. (Ed.): IoTCare 2022, *LNICST* 2023, *501*, pp. 299–308, Springer, Cham, doi: 10.1007/978-3-031-33545-7_21.

[10] J. Beskow, O. Engwall, B. Granström, P. Nordqvist, and P. Wik, "Visualization of speech and audio for hearing impaired persons," *Technology and Disability*, vol. 20, no. 2, pp. 97–107, Jul. 2008, doi: 10.3233/tad-2008-20205.

[11] A. J. Benjamin and K. Siedenburg, "Effects of spectral manipulations of music mixes on musical scene analysis abilities of hearing-impaired listeners," *PLoS ONE*, vol. 20, no. 1, p. e0316442, Jan. 2025, doi: 10.1371/journal.pone.0316442.

[12] R. Da Silva Dessbesel, S. De Carvalho Rutz Da Silva, and E. M. Shimazaki, "Mathematics teaching and Sign Language: a mediated relationship in deaf education," *Revista Internacional De Pesquisa Em Educação Matemática*, vol. 14, no. 5, pp. 1–15, Feb. 2025, doi: 10.37001/ripem.v14i5.3766.

[13] "NV Access." https://www.nvaccess.org/.

[14] "JAWS Screen Reader Overview." https://assistivlabs.com/assistive-tech/screen-readers/jaws.

[15] "Narrator Screen Reader Overview." https://assistivlabs.com/assistive-tech/screen-readers/narrator.

[16] M. Imtiaz, S. Khalid, S. Khadam, S. Arshad, A. Raza, and T. Khalil, "Blind electronic mail system," *Computer and Information Science*, vol. 14, no. 1, p. 14, Jan. 2021, doi: 10.5539/cis.v14n1p14.

[17] B. P. Kumar, M. Sadhik, V. G. Krishna, J. T. Rao, and M. Subhani, "Smart mailing system for blind people," *Zenodo (CERN European Organization for Nuclear Research)*, Feb. 2023, doi: 10.5281/zenodo.7632998.

[18] "Converter for translating text into Braille" https://tiflocentre.ru/calculator-braille-converter.php.

[19] "ABC Braille." https://www.abcbraille.com/braille.

[20] "Braille Translator: ⠠⠞⠑⠭⠞ Text Decoder and Converter (◕‿◕) SYMBL." https://symbl.cc/ru/tools/braille/.

[21] C. Pinheiro, F. Moreira da Silva, "Colour, typography & aged vision – An inclusive design approach," In *12th International AIC Colour Congress, The Sage Gateshead*, 2013, Newcastle.

[22] BGAssist, "Tactonom Reader – Interactive tactile graphic reading device." *BGASSIST LTD*, Dec. 05, 2024. https://shop.bgassist.com/produkt/tactonom-reader.

[23] B. Nethravathi *et al.*, "Visually impaired person assistance based on Tensor FlowLite technology," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 13, no. 11, Jan. 2022, doi: 10.14569/ijacsa.2022.0131171. Available: https://doi.org/10.14569/ijacsa.2022.0131171.

[24] T. Stefanov, S. Varbanova, M. Stefanova, and I. Alinski, "Mobile app prototype supporting people with special needs," *TEM Journal*, pp. 1871–1880, Aug. 2024, doi: 10.18421/tem133-15.

[25] N. Manukyan, P. Hakobyan, A. Harutyunyan, "Analysis of Students' Learning Behavior via Moodle Data," *CSIT Conference 2021 Proceedings*, Yerevan, Armenia, September 27 - October 1. 2021, 278-281.

[26] S. Shafique *et al.*, "SnapStick: Merging AI and accessibility to enhance navigation for blind users," *Technologies*, vol. 13, no. 7, p. 297, Jul. 2025, doi: 10.3390/technologies13070297.

[27] NeuralWriter, "AI Image Generator ➤ Free text-to-image generator | NeuralWriter," *NeuralWriter*. https://neuralwriter.com/image-generator-tool/.

How Teachers' Gestural Culture Influences Japanese Students' Emotions: A Machine Learning Approach

Yuka Nishi[1], Olivia Kennedy[2], Choi Dongeun[3], Noriaki Kuwahara[4]*
Graduate School of Science and Technology, Kyoto Institute of Technology, Kyoto, Japan[1, 3, 4]
Center for Social and Biomedical Engineering, Kyoto Institute of Technology, Kyoto, Japan[4]
Nagahama Institute of Bioscience and Technology, Nagahama, Japan[2]

***Abstract*—This study analyzes differences in teachers' gestural styles based on their culture and investigates how these differences are perceived to influence Japanese students' emotional responses by active observers. Classroom videos of Japanese- and English-native instructors were analyzed using MediaPipe for gesture tracking and DeepFace for facial emotion recognition. Ground-truth emotion labels were collected from four Japanese observers. Results show that Japanese and non-Japanese teachers' gesture dynamics differ in terms of range, rhythm, and symmetry. Japanese student observers perceived each group's gestures differently, with cultural familiarity playing a role in their shifts in emotion. Machine learning models trained on gesture features, facial emotion scores, and teacher background successfully predicted students' affective reactions. These findings highlight the importance of culturally sensitive nonverbal communication in education and demonstrate the potential of AI-based approaches for modeling student emotion in cross-cultural contexts. This study contributes a novel multimodal framework that integrates gesture dynamics, facial emotion recognition, and teacher cultural background to predict student affect, thereby highlighting the necessity of culturally adaptive affective computing in education.**

***Keywords*—Nonverbal behavior; affective computing; AI-based gesture recognition; cultural differences; multimodal analysis; cross-cultural education; emotion prediction**

I. Introduction

Understanding how teachers' nonverbal behavior influences student learning has become increasingly important, particularly in intercultural educational contexts. Among various nonverbal cues, gestures play a pivotal role in directing attention, emphasizing content, and conveying affective meaning. Prior research has shown that gestures not only supplement verbal explanations but also shape learners' cognitive and emotional engagement [1], [2].

However, most of the existing literature on classroom gestures has relied on qualitative observations, which lack precise or scalable measurement approaches [3] [4]. These limitations are especially notable when considering the effect of cultural differences in gesture usage. High- and low-context communication styles, as theorized by Hall [5], may lead to mismatches in how gestures are produced by teachers and interpreted by students. For example, Japanese students may not always correctly interpret gestures used by non-Japanese instructors [6].

Such variation has concrete implications for the classroom. East Asian instructors have been found to favor linking gestures for conceptual clarity, whereas Western educators tend to use more illustrative, expressive forms [7]. These differing gestural styles are not just cultural artifacts but pedagogical strategies that carry affective outcomes. Although the nonverbal behaviors that teachers use reduce psychological distance and have been associated with higher student motivation and engagement [8], such cues may be interpreted differently across cultures.

Beyond cultural and pedagogical dimensions, gestures also serve distinct semiotic functions. McNeill and Kendon classified gestures into types based on how they encode meaning, ranging from spatial reference (deictic) to rhythm and cadence (beat) [2] [4]. These types align with cognitive mechanisms such as mental imagery and discourse structuring [9] [10]. However, their effectiveness depends on how gestures and speech combine to convey meaning in real time [11], and on cultural display rules that modulate students' interpretations [5].

Coinciding with these cultural phenomena, technological advances in affective computing have enabled the quantitative analysis of emotional reactions in real-time classroom environments. Recent advances in AI-based technologies—such as those enabling real-time gesture tracking and facial affect recognition—offer scalable means to detect student emotions in classroom environments [12][13][14]. These methods, however, must account for cultural nuances in gesture production and emotional expression.

This study focuses on how teachers' culturally shaped gestural styles influence Japanese students' emotional perceptions during classroom interactions. Rather than analyzing gesture types per se, we examine how gesture dynamics vary between Japanese and non-Japanese instructors, and how those differences are perceived to affect changes in students' emotions by Japanese observers, simulating a classroom perspective. We also evaluate the extent to which these responses can be predicted using machine learning models trained on gestural and facial features. In other words, the central research question we address is how culturally conditioned differences in teachers' gestures influence Japanese students' affective responses, including the extent to which these responses can be systematically modeled using AI-based analysis.

II. Related Work

Understanding the role of nonverbal teacher behavior in shaping student engagement has been a long-standing point of interest to researchers. Gestures, facial expressions, and body

orientation contribute not only to the transmission of content but also to the affective climate of the classroom. Foundational studies have emphasized the importance of gesture in supporting conceptual understanding and memory formation [1][2]. However, most prior work in this area has relied on observational or qualitative methods [3][4].

Cultural context significantly shapes the form and interpretation of nonverbal behavior. Hall's theory on high- and low-context communication [5], laid the groundwork for understanding why gestures may be interpreted differently across cultures. Park et al. [6] further demonstrated that students from different cultural backgrounds may misinterpret hand gestures used by foreign teachers. Comparative educational studies also show that East Asian teachers tend to use more "linking" gestures to build logical connections, while Western teachers often gesture more illustratively [7].

From a pedagogical standpoint, higher levels of teacher immediacy—the nonverbal behavior used to reduce psychological distance— has been consistently associated with increased student motivation and engagement [8]. These behaviors include open gestures, sustained eye contact, and expressive facial expressions. However, what constitutes "immediacy" may vary by cultural norms, complicating how students interpret these behaviors [5].

Beyond their affective and cultural functions, teacher gestures also have distinct symbolic roles that help impart meaning in instructional discourse. McNeill [2] and Kendon [4] categorized gestures not just by their form, but by how they encode referential or abstract information. In this framework, deictic gestures refer to spatial or discourse objects, iconic gestures depict physical forms, metaphoric gestures give form to abstract ideas via spatial examples, and beat gestures offer rhythm and cadence. These categories align with cognitive processes such as mental imagery, thought activation, and discourse segmentation [9] [10]. In classroom contexts, these functions may influence students both in terms of comprehension and emotional response, especially when the gesture reinforces the accompanying verbal message [11].

In recent years, AI-based techniques have increasingly been applied to classroom research, enabling objective, fine-grained analysis of gesture and emotion. Computer vision systems like MediaPipe and DeepFace allow automatic tracking of hand motion and facial affect, making them useful as new tools for educational affective computing. Whitehill et al. [12] proposed a system to estimate a student's level of engagement via facial cues, finding that machine-measured engagement levels predicted learning outcomes better than prior academic performance. Dai et al. [13] extended this approach by combining multiple modes of tracking—including audio and transcript sentiment—to detect negative teacher emotion (e.g., frustration) across large video datasets. These developments point to the feasibility of real-time, AI-driven feedback systems that support teachers in monitoring classroom sentiment [14].

Although these systems still face challenges in recognizing culturally nuanced gestures or subtle emotions [13], their performance thus far marks a significant step toward quantifying affective dynamics in education. However, few studies explicitly incorporate the teacher's cultural background as a predictive feature when modeling student emotional responses. Combining both multimodal learning analytics and cross-cultural gesture, research enables a more holistic understanding of teacher–student interaction and could lead to change in both pedagogical design and teacher professional development.

While prior studies have examined cultural differences in nonverbal communication [6], few have systematically quantified how gestures differ by teacher cultural background and how those gestures are emotionally perceived by students. This study aims to fill this gap by employing a combination of gesture kinematics, facial emotion analysis, and teacher cultural identity in a unified predictive framework.

Recent advances in AI-based approaches have significantly enhanced the analysis of teachers' nonverbal behaviors and the evaluation of student affect. For instance, Yoon et al. proposed a MediaPipe-based system to quantify pointing gestures by monitoring asynchronous video lectures and demonstrated that in-service teachers adapted their gestures more effectively than pre-service teachers [15]. Similarly, Chen et al. developed a real-time gesture recognition framework using MediaPipe BlazePose to detect teachers' hand signals and deliver attentional cues to students with identified attention deficit problems; doing so achieved an F₁ score of approximately 0.88 [16]. Shou et al. introduced a multi-scale convolutional neural network (CNN) with fine-grained attention enhancements for recognizing student facial expressions in real classroom settings; reporting 93 % accuracy on in-school datasets [17]. Singh et al. gauged online learners' engagement with a multi-modal student attentiveness detection model (MMSAD) that fuses CNN-based facial expression analysis with speech activity detection [18]. Also using CNN-based emotion recognition, Salloum et al. demonstrated pipeline for real-time integration into online learning platforms; achieving 95% accuracy on the FER2013 benchmark [19]. These studies highlight the growing trend of leveraging MediaPipe, DeepFace, and deep learning models to monitor both teacher gestures and student emotions in educational environments. Such AI techniques provide a robust foundation for the work involved with in this study, which employs MediaPipe and DeepFace to perform a cross-cultural analysis of teacher gestures and their impact on Japanese students' affect.

Building on these developments, our study aims to bridge a crucial gap in cross-cultural gesture analysis literature by examining how culturally shaped gestural styles influence students' emotional perceptions in intercultural classrooms. While prior research has demonstrated the utility of AI-based systems for gesture tracking and emotion recognition, few studies have explicitly modeled the cultural background of teachers as a predictive factor in student affect; such cultural context is often treated as peripheral rather than integral to affective dynamics. We address this by integrating gesture kinematics, facial emotion features, and teacher cultural identity into a unified machine learning framework. This allows us to investigate not only behavioral differences across cultures, but also how these differences impact emotional interpretation by students. In doing so, we contribute to the development of affective computing systems that can adapt to culture, in order to better reflect the complexities of human communication in diverse educational settings.

III. Methodology: AI-Based Multimodal Analysis of Culturally Conditioned Gestural Impact

This study uses a multimodal analysis framework to investigate how cultural differences in teachers' gestural behavior influence Japanese students' emotional responses. Fig. 1 illustrates the overall workflow of the study, from video collection to modeling. As shown in Fig. 1, the procedure consists of five main steps:

- Video collection from Japanese and non-Japanese instructors
- Gesture clip extraction
- Observer-based affective labeling
- Feature extraction using MediaPipe and DeepFace
- Statistical analysis and machine learning modeling

Each step is described in detail in the following subsections.

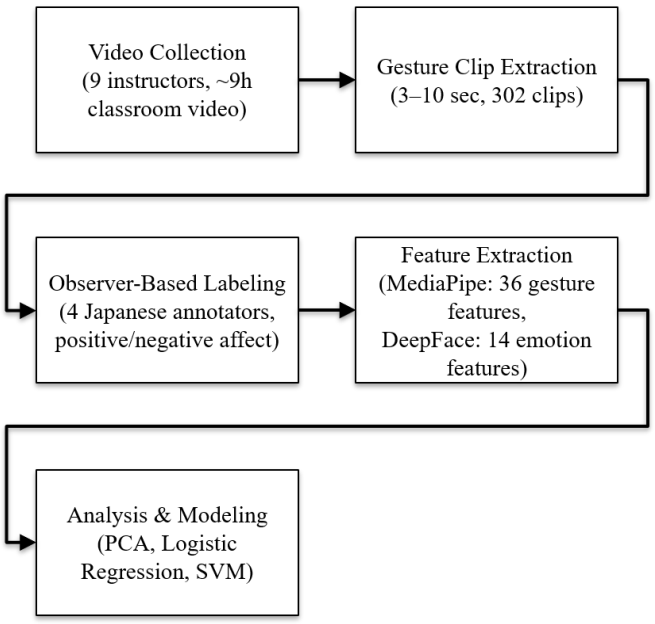


Fig. 1. Overall workflow of the study, from classroom video collection to predictive modeling of student affect.

We employed computer vision tools to extract gesture dynamics, analyzed student affect based on their facial expressions and observer labels, and built machine learning models incorporating both motion features and teacher cultural identity.

A. Dataset

Video data were collected from nine English language instructors (five native Japanese and four native English speakers) teaching Japanese university students in classroom settings. The total length of the recordings was approximately nine hours. The classroom videos were originally recorded as part of a study on spontaneous dynamic teacher emotion recognition by Kennedy et al. [20] and were reused in this study with appropriate ethical approval. Gesture clips of three to ten seconds were segmented from these videos based on visible hand movement, excluding clips where gestures were obstructed or out-of-frame. Only segments where the teacher was facing forward with both hands visible were included. In total, 302 clips were extracted. This dataset was selected because it captures authentic classroom interactions in a naturalistic setting, rather than staged or laboratory recordings, ensuring ecological validity. While the sample size is modest and limited to nine instructors, it provides a balanced comparison between Japanese and non-Japanese teachers in real educational contexts, which is essential for addressing our research questions.

B. Observer-Based Labeling

Four Japanese female annotators, aged 20 to 23, independently labeled each gesture clip by imagining how they, as a student, in the teacher's classroom, might emotionally respond to the teacher's gestures. Their ratings were based on their own perceptions, rather than on observed student reactions. Each annotator viewed the videos in randomized order and was not made aware of the teachers' cultural identities, language backgrounds, and machine-generated emotion scores (positive/negative). Because the focus of this study is gesture, the annotators made their judgments on soundless clips.

The initial consensus label for each clip was determined by majority—when three or more annotators labelled a clip the same way, their shared label was used. In cases of a tie (2 vs. 2), a decision tree model was trained on the majority-agreement cases to predict the most likely final label based on the pattern of individual annotator inputs. This model served to resolve ambiguous labeling outcomes in a reproducible and data-driven manner.

To ensure the validity of this modeling approach, we evaluated the decision tree model's ability to replicate majority-agreement labels through two complementary methods:

- Distribution-level comparison: We applied a chi-squared test and computed Kullback–Leibler divergence to compare the distribution of labels produced by the decision tree model to that of the majority-agreement labels.
- Sample-level comparison (two-two-split cases): We calculated prediction accuracy and Cohen's kappa between the decision tree model's predicted labels and the final consensus labels.

These evaluation methods were used to confirm that the decision tree model reliably approximated consensus labeling in tie situations. All final labels—whether decided by majority or through the decision tree—were then used as ground truth in subsequent supervised learning tasks, as described in Sections *E.2* and *E.3*.

C. Feature Extraction from Gestures

To quantify each teacher's gesture dynamics, we used MediaPipe Holistic (Zhang et al. [21]), which provides real-time tracking of body pose and hand landmarks. All features were computed relative to the teacher's nose landmark to normalize for their placement and the camera's framing. Below is a description of each feature group (totaling 36 dimensions):

1) Wrist Position Statistics (Left and Right): For each wrist, MediaPipe Holistic tracks the trajectory of its (x, y) coordinates

over time. From these trajectories, four descriptive statistics for the x-axis and four for the y-axis were computed—namely, the maximum, minimum, mean, and variance of the wrist position across all frames. These statistics capture the spatial extent (range), central tendency (average position), and variability (how much the hand moves) during the gesture.

2) Hand velocity statistics: Velocity was calculated by differentiating frame-by-frame wrist positions in both *x* and *y* directions. For each axis, we then derived the maximum, minimum, mean, and variance of these instantaneous velocities across the clip. These features reflect how quickly and consistently the teacher moves their hands. High maximum velocity, for example, indicates a fast gesture, while high variance suggests fluctuating speed.

3) Hand acceleration statistics: Acceleration was calculated by taking the temporal derivative of the velocity values along each axis. Like velocity, *x* and *y* accelerations were summarized using their maximum, minimum, mean, and variance across the clip. Acceleration metrics characterize the abruptness or smoothness of movements—large peaks in acceleration denote sudden starts or stops, whereas low variance implies a more uniform movement.

4) Positional correlations: To capture directional consistency and shape of motion, we computed the Pearson correlation coefficient between the *x* and *y* coordinates of each wrist throughout the gesture. A strong positive or negative correlation indicates that the wrists' motions are primarily diagonal, whereas a correlation near zero suggests more independent horizontal and vertical movement.

5) Velocity autocorrelations: Finally, we calculated autocorrelation values for the *x* and *y* components of velocity over short time lags. These correlations measure how similar the hand's speed is from one moment to the next, providing insight into rhythmic or repetitive motion patterns.

In total, these five groups produce a motion profile of 36 numerical features per gesture clip (8 wrist position statistics for each wrist=16, 8 hand velocity statistics=8, 8 hand acceleration statistics=8, 2 positional correlations=2, and 2 velocity autocorrelations=2). Each feature was normalized by clip duration and teacher height before summarization.

While other frameworks, such as OpenPose, have also been widely used for gesture tracking, MediaPipe Holistic was selected because it offers a lightweight and real-time implementation that is less computationally demanding while maintaining comparable accuracy in classroom environments.

This 36-dimensional motion profile, based on MediaPipe Holistic [21], serves as the basis for all subsequent analyses and predictive modeling.

D. Emotion Annotation and Predictive Modeling

This section describes how teacher gestures were analyzed to predict Japanese students' emotional responses, and how gesture characteristics differed systematically between Japanese and non-Japanese instructors. The analysis follows two complementary paths: 1) identifying cultural differences in gesture execution using machine learning, and 2) modeling students' affective responses.

1) Cultural style classification of teachers: To explore how gesture dynamics differ by cultural background, we trained a supervised classifier to distinguish between Japanese and non-Japanese teachers based on the gesture features extracted in Section *C*. This classification task served to identify which motion patterns (e.g., gesture range, rhythm, directional symmetry) are characteristic of each teacher group. The results of this analysis provide an empirical basis for comparing the gestural tendencies of different teacher populations.

2) Observer-based emotion annotation: Japanese students' emotional responses to each gesture were annotated using the four-person consensus protocol, described in Section *B*. Each clip was labeled as eliciting either a positive or a negative emotional reaction.

These consensus labels were then used as ground truth for evaluating whether machine learning models could accurately predict observer-inferred student affective responses based on gesture perception.

3) Prediction of student emotion from gesture features: We first trained a machine learning model to predict the binary emotion label (positive/negative) using gesture features alone. This allowed the model to assess whether the movement pattern of the teacher's gesture, independent of facial cues, was sufficient to explain how Japanese students emotionally responded.

Next, we trained a second model by incorporating both gesture features and facial emotion predictions from DeepFace [22]. DeepFace is a lightweight CNN-based emotion recognition framework capable of estimating seven basic emotion categories (happy, sad, fear, angry, disgust, surprise, neutral) from facial expressions in real-time. It was used to extract per-clip emotion probabilities, which were then averaged over time and used as input features alongside gesture motion. The goal in using this second model was to determine whether combining gesture dynamics with machine-estimated facial affect signals improves prediction performance.

4) Application to feedback and cultural interpretation: These predictive models can be used to provide actionable feedback to teachers by identifying gesture patterns that are more likely to be perceived negatively by students. Such insights can guide the creation of culturally responsive teaching strategies and help educators avoid nonverbal behaviors that may unintentionally cause discomfort or disengagement.

Likewise, the analysis of model coefficients and misclassified cases can help us understand how specific gestural differences between Japanese and non-Japanese teachers contribute to affective outcomes. By systematically comparing model behavior across teacher groups, we aim to identify which motion traits are positively or negatively received in a Japanese cultural context.

E. Implementation

All processing was performed in Python using MediaPipe, DeepFace, and scikit-learn. For gesture-based cultural classification (i.e., distinguishing between Japanese and non-Japanese teachers), we used both Support Vector Machines (SVM) and Linear Discriminant Analysis (LDA) to compare performance in modeling stylistic differences. For emotion prediction tasks, logistic regression was primarily used for its transparency, and in some cases, SVM was also applied to assess potential nonlinear relationships.

Although more complex deep learning models could potentially achieve higher predictive accuracy, we deliberately employed logistic regression and SVM because they are interpretable, less prone to overfitting with relatively small datasets, and allow clearer examination of how gesture and facial features contribute to cultural differences.

All of these features were standardized, and stratified cross-validation was applied throughout their application. No extensive hyperparameter tuning was conducted, and default settings and L2 regularization were used as appropriate.

IV. Results

This section presents the empirical findings in four stages. First, the reliability of emotion labels generated through observer annotation and machine-aided resolution are evaluated. Second, cultural differences in teachers' gestural dynamics based on motion features extracted from clips of classroom videos are analyzed. Third, how students' emotional reactions vary depending on the cultural background of the teacher are examined. Finally, machine learning models to predict students' affective responses from gestural and facial features are discussed and the models' performance across different input modalities and teacher groups compared.

A. Reliability of Emotion Labels

To ensure the validity of emotion labels used as ground truth for supervised learning, a multi-step evaluation of observer-labeled data was implemented, as outlined in Section III *B*.

While a majority-vote protocol was used to assign labels, in 30.8% of the samples (72 out of 234 clips), the labels were evenly split (2 vs. 2). To resolve these ambiguous cases, a logistic regression model was trained using the individual ratings from the 162 majority-agreement clips as input and their consensus labels as targets. The trained model was then applied to predict the most probable label in the tie cases.

To assess the validity of this model-based resolution strategy, we conducted two evaluations:

1) Distribution-level consistency: A chi-squared test comparing the emotion label distribution (positive vs. negative) in the predicted set versus the original majority-agreement set showed no significant difference ($p > 0.05$). Additionally, the Kullback–Leibler (KL) divergence between the two distributions was extremely low (0.012), indicating near-identical global distributions.

2) Sample-level agreement: To validate the logistic model's predictive power, we performed five-fold cross-validation on the majority-agreement cases. The model achieved an average accuracy of 95.4% and a Cohen's kappa score of 0.88, demonstrating high consistency with human consensus even in ambiguous cases.

These results suggest that the predicted labels for 2–2 split cases are statistically and behaviorally consistent with majority-vote annotations. All subsequent analyses, including classification and predictive modeling, used this unified set of final labels.

B. Cultural Differences in Gesture Dynamics in Gestures and Emotions

To investigate how nonverbal behavior and emotional expression differ between Japanese (J) and non-Japanese (N) teachers, we conducted a principal component analysis (PCA) on a set of 50 combined dimensions. This set consisted of 36 motion-related features derived from teachers' hand gestures and 14 facial emotion features extracted using DeepFace. All features were standardized before analysis. Component scores were then compared between cultural groups using Welch's t-test, applying Bonferroni correction to control for multiple comparisons.

1) PCA and explained variance: A total of 234 video clips were analyzed, each corresponding to either a Japanese teacher (J) or a non-Japanese teacher (N). For each clip, 50 features were extracted, consisting of:

a) 36 gesture features: These include the maximum, minimum, mean, and variance of left and right wrist positions and velocities in both X and Y directions, as well as wrist velocity correlations.

b) 14 affective features: These represent mean and variance of basic emotional expressions (anger, disgust, fear, happy, sad, surprise, neutral), extracted via facial analysis.

No missing values or filtering was applied. All features were standardized to zero mean and unit variance using StandardScaler from scikit-learn. Principal component analysis (PCA) was conducted to reduce dimensionality and explore patterns associated with cultural differences.

The PCA revealed that the first 14 components together accounted for 81.00% of the total variance in the feature space, with the first component alone capturing 23.2%. Table I shows the explained variance and cumulative variance for each component up to PC14.

2) Statistical comparison: Japanese vs. Native Teachers. We tested for cultural differences in each of the first 14 components using Welch's t-test. Table II presents the p-values (uncorrected and Bonferroni-adjusted) and effect sizes (Cohen's d) for each component. Three components—PC2, PC3, and PC9—showed statistically significant group differences after correction. Adjusted p-values were computed using Bonferroni correction. Values in bold indicate statistical significance (adjusted $p < 0.05$).

TABLE I. EXPLAINED VARIANCE OF PRINCIPAL COMPONENTS (UP TO 80%)

Principal Component	Explained Variance Ratio (%)	Cumulative Variance (%)
PC1	23.2	23.2
PC2	9.1	32.3
PC3	6.9	39.2
PC4	5.9	45.1
PC5	5.5	50.6
PC6	5.0	55.6
PC7	4.3	59.9
PC8	3.7	63.6
PC9	3.3	66.9
PC10	3.3	70.2
PC11	3.1	73.3
PC12	3.0	76.3
PC13	2.4	78.7
PC14	2.3	81

TABLE II. STATISTICAL COMPARISON OF PRINCIPAL COMPONENT SCORES BETWEEN JAPANESE AND NON-JAPANESE TEACHERS

Principal Component	p_value	adjusted_p	Cohen_d
PC1	0.206	>0.05	-0.16
PC2	**<0.001**	**0.001**	**0.52**
PC3	**<0.001**	**<0.001**	**0.82**
PC4	0.534	>0.05	-0.08
PC5	0.137	>0.05	-0.2
PC6	0.1	>0.05	-0.23
PC7	0.506	>0.05	0.09
PC8	0.057	>0.05	0.25
PC9	**<0.001**	**<0.001**	**0.6**
PC10	0.458	>0.05	-0.1
PC11	0.582	>0.05	-0.07
PC12	0.009	>0.05	-0.36
PC13	0.02	>0.05	0.31
PC14	0.732	>0.05	-0.05

1) Interpretation of significant components: We examined the five features with the highest absolute loadings for each significant component (PC2, PC3, PC9), as shown in Table III. Feature names correspond to variables such as RIGHT_WRIST_x_mean (i.e., a mean of the x-position of the right wrist) and velocity_x_correlation (i.e., the correlation coefficient between x-velocities of left and right wrists), etc. "Loading" denotes the PCA factor loading, representing how strongly each feature influences the component.

- PC2: Horizontal motion and facial fear: PC2 differentiates teachers along horizontal gestural dimensions and facial fear variability. Non-Japanese teachers tend to exhibit greater horizontal range and smoother motion in the x-axis (e.g., higher RIGHT_WRIST_x_mean and velocity_x_correlation), while Japanese teachers show higher variation in "fear" expression.
- PC3: Vertical Gesture Amplitude and Negative Affect: PC3 captures vertical gesture positioning and fluctuations in negative emotion. Non-Japanese teachers use broader vertical ranges and faster movements (higher RIGHT_WRIST_y_mean and Vy_mean), while Japanese teachers show greater variance in "sad" facial expression.
- PC9: Postural Stability and Positive Affect: PC9 reflects differences in vertical consistency and "happy" expression. Non-Japanese teachers show more stable hand elevation (higher RIGHT_WRIST_y_mean, lower position_y_correlation) and scored higher in both average and variability of "happy" expressions. Japanese teachers show more fluctuation in vertical posture.

TABLE III. TOP 5 LOADINGS OF SIGNIFICANT PCS

Principal Component	Feature	Loading
PC2	RIGHT_WRIST_x_mean	0.29
	velocity_x_correlation	0.28
	RIGHT_WRIST_x_max	0.28
	LEFT_WRIST_x_mean	-0.27
	fear_variance	-0.26
PC3	RIGHT_WRIST_y_mean	0.29
	RIGHT_WRIST_y_min	0.26
	sad_variance	-0.26
	angry_mean	0.25
	RIGHT_WRIST_Vy_mean	0.23
PC9	position_y_correlation	-0.31
	RIGHT_WRIST_y_mean	0.31
	LEFT_WRIST_y_variance	0.28
	happy_variance	0.28
	happy_mean	0.27

2) Interpretation and cultural implications: The PCA results reveal meaningful cultural contrasts in teachers' nonverbal and emotional behavior. Among the 14 principal components which, when combined, explained 81.0% of the variance, three components—PC2, PC3, and PC9—exhibited significant differences between Japanese and non-Japanese instructors.

- PC2 reflected differences in horizontal hand motion and fear-related facial variability; non-Japanese teachers showed greater lateral gesture range and motion consistency, whereas Japanese teachers exhibited more fluctuation in fear expression.
- PC3 involved vertical hand amplitude and negative affect; non-Japanese instructors used broader, faster

vertical gestures and expressed more "anger", while Japanese teachers showed greater variability in "sad" expression.

- PC9 was associated with postural stability and positive facial affect; non-Japanese teachers maintained higher and more consistent hand elevation with more pronounced "happy" expressions, whereas Japanese teachers exhibited greater variability in both gesture and affect.

These findings support the general view that low-context communicators (e.g., non-Japanese teachers) tend to favor explicit and clear nonverbal behavior [2] [5], whereas high-context communicators (e.g., Japanese teachers) rely more on subtle and context-sensitive expression [23]. The combination of gesture and facial features in this analysis allows us to better understand how teachers from different cultural backgrounds encode emotional signals during instruction.

A more detailed interpretation of these patterns—especially in relation to cultural display rules, teacher immediacy, and affective computing design—is provided in the Discussion (see Section V).

C. Student Emotion by Teacher Culture and Its Predictive Model

1) Group differences in student emotion: To investigate whether a teacher's culture influences their students' emotional responses, we compared the distribution of observer emotion labels (positive vs. negative) across teacher cultural backgrounds (Japanese or non-Japanese). A 2×2 contingency table was constructed (see Table IV), and both the chi-squared test and Fisher's exact test were performed.

TABLE IV. CROSS-TABULATION OF TEACHER CULTURE (JAPANESE VS. NON-JAPANESE) AND STUDENT EMOTION LABEL (POSITIVE VS. NEGATIVE)

Teacher Culture	Student Emotion: Positive	Student Emotion: Negative
Japanese (J)	58	45
Non-Japanese (N)	56	75

The results of the chi-squared test approached significance ($\chi^2 = 3.72$, $p = 0.054$, $df = 1$), and Fisher's exact test yielded a statistically significant result ($p = 0.048$, odds ratio = 1.73).

Although non-Japanese teachers are often seen as more expressive, here Japanese teachers more frequently elicited positive emotional responses (56.3% vs. 42.7%).

This suggests that cultural familiarity may take precedence over expressiveness in shaping students' emotional receptivity. To investigate whether Japanese students' emotional responses can be predicted from teachers' nonverbal behavior, we trained logistic regression model using 50-dimensional features derived from 36 gesture dynamics and 14 facial expressions (see Section IV *B*). Cultural labels were excluded from the models to focus solely on behavioral cues. Using five-fold stratified cross-validation, the model achieved a mean accuracy of 66.2% (σ = 7.5%), suggesting that nonverbal behavior alone provides moderate predictive power. As shown in Table V, however, accuracy differed substantially by teacher culture: 62% for Japanese teachers versus 76% for non-Japanese teachers.

TABLE V. ACCURACY OF LOGISTIC REGRESSION MODELS PREDICTING STUDENT EMOTION, BROKEN DOWN BY TEACHER CULTURE (JAPANESE VS. NON-JAPANESE)

Teacher Culture	Accuracy Mean	Accuracy Std
Japanese	0.62	0.061
Non-Japanese	0.76	0.090

To understand the source of this discrepancy, we compared model coefficients between two cultural groups. Table VI lists the five features showing the greatest differences. For non-Japanese teachers, positive student responses were more strongly associated with high hand velocity (e.g., RIGHT_WRIST_Vx_min, LEFT_WRIST_Vx_max) and consistent vertical motion (position_y_correlation). Meanwhile, for Japanese teachers, positional extremes such as RIGHT_WRIST_x_min played a more prominent role in positive student responses—possibly reflecting a culturally embedded preference for subtle, controlled gestures. These findings highlight the culturally situated nature of affective communication and underscore the value of adapting emotion-prediction models to cultural context.

TABLE VI. FIVE FEATURES WITH THE LARGEST COEFFICIENT DIFFERENCES BETWEEN LOGISTIC REGRESSION MODELS TRAINED ON JAPANESE-TEACHER DATA (J) AND NON-JAPANESE-TEACHER DATA (N)

Feature	J_Coefficient	N_Coefficient	Coefficient Absolute Difference
RIGHT_WRIST_Vx_min	0.82	-1.11	1.93
RIGHT_WRIST_x_min	-1.19	0.25	1.44
LEFT_WRIST_Vx_max	0.60	-0.79	1.39
position_y_correlation	0.37	-0.80	1.18
RIGHT_WRIST_x_mean	1.26	0.25	1.01

2) Summary and cultural reflection: The findings in this section (see also Table V) demonstrate that the cultural background of the teacher not only influences the distribution of students' emotional responses but also modulates the internal structure of the predictive model. Logistic regression classifiers, which are trained separately on Japanese and non-Japanese teachers' data, revealed notable differences in accuracy: student emotion was predicted with significantly higher precision when the teacher was a non-Japanese speaker. This suggests that the affective cues conveyed by native teachers were more consistently interpreted by Japanese students, leading to more stable input–output mappings in the model.

Interestingly, Japanese teachers—who are often described as emotionally reserved in high-context cultures [5] [23]—elicited less predictable emotional responses as perceived from a student's perspective. And yet, this does not imply that their gestures were affectively neutral or irrelevant. Rather, lower

gesture predictability may reflect greater contextual modulation or individual variation in how their nonverbal behaviors are perceived. In fact, this aligns with our findings in Section IV *B.2*, where Japanese teachers exhibited richer vertical gestural amplitude and a broader affective range than might be expected from cultural stereotypes.

This suggests that interactions with culturally dissimilar instructors—such as non-Japanese teachers—may enable otherwise restrained Japanese students to externalize their emotional responses more visibly. And so, our model not only predicts emotion, but also reveals how cultural contrast may enhance how well we perceive student affect in educational interactions. This is a particularly important consideration in intercultural classrooms, where emotional communication may otherwise remain latent.

Taken together, these findings indicate that predictive models must be both behaviorally precise and also culturally responsive. While non-Japanese teachers exhibited more overt and consistently interpreted nonverbal-affective behaviors, Japanese teachers demonstrated context-sensitive expressiveness that was no less impactful, though it may vary more. Overall, these results imply that culturally sensitive affective computing models must account for not only static behavioral norms but also dynamic pedagogical roles and expectations.

V. Discussion

This study sheds light on the complex interplay between cultural background, nonverbal behavior, and affective perception in classroom communication. The concept of nonverbal leakage—concealed emotions and intentions "leaking" through subtle facial and gestural cues [24]—helps explain why even small differences in teacher expressiveness can significantly affect student emotion. Building on prior studies in gesture typology [2][4], immediacy theory [8], and cultural context models [5][23], we quantitatively examined how gesture dynamics and facial affect from teachers of different cultural backgrounds influence Japanese students' emotional responses.

Although the emotion annotations in this study were not derived from actual student reactions, they were produced by native Japanese observers explicitly instructed to adopt the perspective of a student in the classroom. This imagined-student approach allowed us to assess affective responses to teacher gestures in a controlled and consistent manner. Prior work in affective computing and cross-cultural psychology supports the use of projected or observer-based affect labeling when direct feedback is impractical or ethically constrained.

Moreover, recent AI-based works such as quantifying gestures via MediaPipe [15], detecting real-time hand signals using BlazePose [16], and high accuracy recognition of student expressions [17][18]–demonstrates how automated, multimodal feedback can guide instructors. These advances support our use of MediaPipe and DeepFace, as well as point toward future systems that offer real-time, culturally sensitive classroom guidance.

A. Cultural Differences in Nonverbal Expressivity

Our findings confirm that cultural background shapes not only how teachers' gesture, but also how those gestures are emotionally interpreted. PCA results revealed that non-Japanese teachers tended to use gestures with broader horizontal and vertical range, smoother trajectories, and more expressive facial signals. On the other hand, Japanese teachers showed more localized motion patterns and higher variability in facial expressions such as sadness and fear. These patterns are consistent with Hall's theory that high-context cultures favor more subtle, less explicit communication styles [5].

Interestingly, although non-Japanese teachers demonstrated more clear and systematic nonverbal behavior, Japanese teachers were more likely to elicit positive affective responses (56.3% versus 42.7%). This suggests that cultural familiarity may play a stronger role in emotional receptivity than expressive intensity alone—a finding that challenges the assumptions derived from immediacy theory [8].

B. Predictive Modeling and Cultural Modulation

Logistic regression classifiers trained on gesture and affective features showed moderate accuracy in predicting student emotion (66.2%). However, when trained separately by teacher group, model accuracy was significantly higher for non-Japanese teachers (72.3%) than for Japanese teachers (64.8%). This suggests that non-Japanese teachers' affective cues were more consistently understood by students, whereas Japanese teachers' cues exhibited greater contextual or individual variation, possibly due to more nuanced display rules [23].

The top differentiating features between the two models—including RIGHT_WRIST_Vx_min, position_y_correlation, and RIGHT_WRIST_x_min—highlight how affective perception hinges on both intensity and spatial/rhythmic regularity. These quantitative findings resonate with earlier symbolic accounts of gesture [10] [11] that emphasize cognitive alignment and expressivity.

C. Cultural Contrast and Student Legibility

One particularly compelling implication is that interactions with culturally dissimilar instructors may enhance the visibility of student emotion. When students are exposed to low-context, overtly expressive behavior—typical of non-Japanese teachers—they may externalize their own emotional states more clearly. This could explain the more stable prediction accuracy for these cases, which would suggest that cultural dissimilarity may sometimes reduce ambiguity in affective signaling.

Conversely, Japanese teachers, often operating within a high-context communicative norm, may trigger more individualized and internalized affective interpretations, complicating the ability to predict affect. However, this should not be mistaken for lower emotional engagement. On the contrary, the data indicate that Japanese teachers express a broader emotional range, but with less behavioral regularity. This calls for more context-sensitive modeling approaches.

D. Implications for Culturally Adaptive Affective Computing

The implications of these findings extend beyond education to the broader field of affective computing. As previous literature has emphasized [12] [13], multimodal AI models must

capture both real-time behavior and contextual nuance. The results here suggest that affective computing systems should explicitly model cultural context—not only as a background variable but as a dynamic factor that modulates both expression and perception.

Future systems should adapt to the emotional norms, gestural practices, and display rules of the cultural environment in which they are used. This would allow such systems to move beyond accuracy benchmarks and toward human-aligned interpretability, which would then enhance trust and engagement in human–AI interaction across diverse populations.

VI. Limitations and Future Work

While this study offers novel insights into the relationship between teachers' nonverbal behavior and student emotion in cross-cultural classrooms, several limitations must be acknowledged.

First, the analysis was based on 234 video clips, selected from an initial pool of 302 after excluding segments that were too short, obstructed, or poorly framed. Although this filtering ensured data quality and consistency, the dataset was relatively small and drawn from a limited number of instructors—five Japanese and four non-Japanese teachers. This sample size restricts the generalizability of the findings, as it may not fully capture the diversity of instructional styles or cultural expression within each group. Future research should aim to expand both the number and variety of participating educators.

Second, emotional annotations were limited to a binary emotional scale (positive vs. negative), which simplifies the complexity of affective responses. Additionally, these annotations were not based on actual student reactions but were provided by four Japanese university students who imagined themselves as learners in the classroom. While this approach ensured cultural and procedural consistency, it introduces the possibility of projection bias and limits ecological validity. Future studies should consider using a more diverse annotation pool and explore the use of continuous or multidimensional emotion metrics.

Third, although facial expressions and gesture dynamics were combined in the predictive model, the analysis was conducted at the clip level, without fine-grained time sync between gestures and emotional shifts. Sub-clip or frame-level modeling might reveal more precise temporal cues associated with emotional perception, particularly in the case of micro-expressions or beat gestures.

Fourth, cultural differences were inferred solely from the teachers' identities (Japanese vs. non-Japanese), without explicitly modeling student cultural frameworks or individual differences. While this study assumed cultural homogeneity among students—all of whom were Japanese—the model did not incorporate any student-level variables, such as prior experience with foreign instructors, which may influence how nonverbal cues are perceived.

Fifth, logistic regression was chosen for its interpretability, but it may not capture more complex, nonlinear relationships between multimodal features and affective outcomes. Although suitable for initial exploration, future studies could employ deep learning or transformer-based architectures.

Finally, we acknowledge that, although their evaluations were consistent and culturally grounded, the homogeneity in gender and age of the emotion annotators in this study may introduce bias in emotion perception. Prior literature suggests that emotional sensitivity and interpretation can vary across genders. To enhance the robustness and generalizability of affective labeling, future studies should incorporate a more demographically diverse annotator pool.

Future research should explore more advanced AI architectures to overcome these limitations. For example, transformer-based multimodal models could capture temporal dependencies between teacher gestures and student facial expressions in finer detail. Additionally, evaluating AI-based affect estimation methods in online or hybrid learning environments would help generalize findings beyond face-to-face classrooms. Finally, incorporating student-level variables—such as prior exposure to foreign instructors—into predictive models could enable personalized, culturally adaptive feedback systems for real-time instructional support.

Taken together, these limitations point to several directions for future research. Expanding sample diversity—both among teachers and student annotators—will improve the generalizability of findings, while refining emotional representations through continuous or multidimensional metrics can capture subtler affective nuances. Enhancing temporal resolution to the frame level and incorporating student-level variables (e.g., gender, prior cross-cultural experience) will yield deeper insight into how culture shapes affective dynamics. Moreover, exploring teacher–student emotional co-regulation models in real-time classroom interactions could illuminate bidirectional affective processes. Such advances will not only support the development of adaptive, culturally responsive feedback systems but also enhance student engagement and emotional well-being in diverse educational settings.

VII. Conclusion

This study examined how differences in teachers' gestures in classroom settings based on their culture influence Japanese student observer's emotional responses. By combining gesture kinematics and facial emotion analysis with machine learning techniques, both the form and cultural context of nonverbal behavior were demonstrated to have a significant influence on affective perception and predictability.

While non-Japanese teachers were found to use more overt gestures that were easier for systems to interpret, Japanese teachers were more likely to elicit positive emotional responses, which highlights the role of cultural familiarity in affective engagement. Interestingly, the findings also suggest that cultural contrast may enhance the ability to read student affect. This raises important questions about how intercultural dynamics shape not only teacher expressivity but also student emotional visibility in classroom interactions.

Predictive modeling further revealed that non-Japanese instructors yielded higher emotion prediction accuracy, likely due to greater behavioral regularity, while Japanese teachers' gestures exhibited more contextual variability. This variability

was further reflected in the predictive models used, which achieved higher prediction accuracy for non-Japanese teachers. These differences underscore the need for culturally adaptive models that accommodate variation in signal consistency and contextual modulation.

These findings emphasize that differences in student perception are not merely a matter of gesture intensity or clarity, but are shaped by culturally embedded expectations, display rules, and interaction norms. By quantifying these dynamics and modeling affective responses, this study demonstrates that affective computing for education must be both multimodal and culturally adaptive.

By providing a scalable, quantitative framework for analyzing cross-cultural emotion perception in classrooms, this study contributes not only to research on gesture and education but also to the design of AI systems, that can be used to monitor human emotion. Future studies should build on these findings by incorporating student-level diversity, more discrete emotion labels, and real-time feedback mechanisms to support more inclusive and emotionally intelligent educational environments.

REFERENCES

[1] S. Goldin-Meadow, *Hearing Gesture: How Our Hands Help Us Think*, Cambridge, MA: Harvard University Press, 2003.

[2] D. McNeill, *Hand and Mind: What Gestures Reveal about Thought*, Chicago, IL: University of Chicago Press, 1992.

[3] A. B. Hostetter and M. W. Alibali, "Visible embodiment: Gestures as simulated action," *Psychon. Bull. Rev.*, vol. 15, no. 3, pp. 495–514, 2008.

[4] A. Kendon, *Gesture: Visible Action as Utterance*, Cambridge, U.K.: Cambridge University Press, 2004.

[5] E. T. Hall, *Beyond Culture*, New York: Anchor Books, 1976.

[6] J. Park, M. Valstar, and M. Pantic, "Cultural differences in nonverbal communication: A study of manual gestures," *IEEE Trans. Affective Comput.*, vol. 3, no. 2, pp. 134–147, 2012.

[7] G. S. Lee and A. K. Ng, "Linking gestures and cognition: A cross-cultural analysis of math classrooms," *Int. J. Educ. Res.*, vol. 98, pp. 1–14, 2019.

[8] A. L. Witt and R. Schrodt, "Teacher immediacy and student learning: A meta-analytic review," *Commun. Educ.*, vol. 55, no. 1, pp. 1–22, 2006.

[9] C. Müller, "Forms and uses of the palm up open hand: A case of a gesture family?," *Gesture*, vol. 1, no. 2, pp. 149–171, 2001.

[10] A. Cienki and C. Müller, Eds., *Metaphor and Gesture*, Amsterdam: John Benjamins, 2008.

[11] J. Mittelberg, "Gesture and speech from a semiotic perspective: A fresh look at the relationship between mimetic gestures and language," *Gesture*, vol. 9, no. 1, pp. 1–28, 2009.

[12] J. Whitehill, Z. Serpell, Y. Lin, A. Foster, and J. R. Movellan, "The Faces of Engagement: Automatic Recognition of Student Engagement from Facial Expressions," *IEEE Trans. Affective Comput.*, vol. 5, no. 1, pp. 86–98, 2014.

[13] W. Dai, C. Heffernan, and J. Davis, "Detecting Negative Teacher Affect in the Classroom Using Multimodal Signals," *Proc. Int. Conf. Learning Analytics & Knowledge (LAK)*, pp. 231–240, 2023.

[14] K. D'Mello and S. Graesser, "AutoTutor and affective learning," in *New Perspectives on Affect and Learning Technologies*, Springer, 2011, pp. 35–48.

[15] H. Y. Yoon, S. Kang, and S. Kim, "A non-verbal teaching behaviour analysis for improving pointing out gestures: The case of asynchronous video lecture analysis using deep learning," *J. Comput. Assist. Learn.*, vol. 40, no. 3, Aug. 2024.

[16] I. D. S. Chen, C. M. Yang, S. S. Wu, *et al.*, "Continuous recognition of teachers' hand signals for students with attention deficits," *Algorithms*, vol. 17, no. 7, Art. no. 300, Jul. 2024.

[17] Z. Shou, Y. Huang, D. Li, *et al*., "A student facial expression recognition model based on multi-scale and deep fine-grained feature attention enhancement," *Sensors*, vol. 24, no. 20, Art. no. 6748, Oct. 2024.

[18] R. Singh, E. Ramanujam, and M. N. Babu, "MMSAD – A multi-modal student attentiveness detection in smart education using facial features and landmarks," *J. Ambient Intell. Smart Environ.*, vol. 15, no. 1, Jan. 2025.

[19] S. A. Salloum, K. M. Alomari, A. M. Alfaisal, R. A. Aljanada, and A. Basiouni, "Emotion recognition for enhanced learning: Using AI to detect students' emotions and adjust teaching methods," *Smart Learn. Environ.*, vol. 12, Art. no. 21, Mar. 2025.

[20] O. Kennedy, N. Kuwahara, T. Noble, and C. Fukada, "Accuracy of Spontaneous Dynamic Teacher Emotion Recognition in Japanese College Students," under review, *Frontiers in Education*, 2025.

[21] F. Zhang, V. Bazarevsky, A. Vakunov, et al., "MediaPipe Hands: On-device real-time hand tracking," *arXiv preprint arXiv:2006.10214*, 2020.

[22] S. Serengil and A. Ozpinar, "LightFace: A hybrid deep face recognition framework," *Data Science and Applications*, vol. 1, no. 1, pp. 1–8, 2021.

[23] D. Matsumoto, "Cultural similarities and differences in display rules," *Motivation and Emotion*, vol. 14, no. 3, pp. 195–214, 1990.

[24] P. Ekman and W. V. Friesen, "Nonverbal leakage and clues to deception," *Psychiatry*, vol. 32, no. 1, pp. 88–106, 1969.

Enhancing Approximate Conformance Checking Accuracy with Hierarchical Clustering Model Behaviour Sampling

Yilin Lyu
School of Computing and Information System,
The University of Melbourne, Parkville, Victoria 3010, Australia

***Abstract*—Conformance checking techniques evaluate how well a process model aligns with an actual event log. Existing methods, which are based on optimal trace alignment, are computationally intensive. To improve efficiency, a model sampling method has been proposed to construct a subset of model behaviour that represents the entire model. However, current model sampling techniques often lack sufficient model representativeness, limiting their potential to achieve optimal approximation accuracy. This study proposes new model behaviour sampling approaches using hierarchical clustering to compute an approximation closer to the exact result. This study also refines the existing upper bound algorithm for better approximation. Our experiments on six real-world event logs demonstrate that our method improves approximation accuracy compared to state-of-the-art model sampling methods.**

***Keywords*—Approximate conformance checking; model behaviour sampling; hierarchical clustering; process mining**

I. Introduction

Conformance checking is a set of process mining functionalities aimed at identifying deviations between the actual behaviour of the event log ("as-is") and the modeled behaviour of the process model ("to-be"). It facilitates further applications, such as model repair, anomaly detection, and algorithm evaluation [1]. In recent years, the alignment-based method [2] has become the de facto standard for conformance checking in the computation of conformance diagnostics, since it always returns the most accurate deviations, known as optimal alignment [3]. However, finding the optimal alignment is an NP-hard problem [4]. As the complexity of the log and model increases, the run-time complexity of optimal alignment computation grows exponentially, leading to extremely long computation times, sometimes even taking several weeks. This makes them impractical for real-world applications, especially large-scale event logs. Moreover, in certain cases, an exact conformance value is not necessary, such as when performing a preliminary evaluation of process models with various process discovery algorithms [5].

To address the problems, various approximation strategies have been proposed, including optimizing the search algorithm [6], [7] and decomposition schemes [8], [9]. However, sampling provides another angle for approximate conformance checking, such as sampling traces to represent the event log [10], [11] or selecting model traces to substitute for the process model [5], [12]. In this study, we adopt the latter approach, focusing on model sampling. Two main model sampling methods exist: simulation [13] and candidate selection [5]. We concentrate on candidate selection due to its higher accuracy [5]. The candidate selection method identifies representative traces from the event log (i.e., log behaviour subset), and then computes their optimal alignments to determine the corresponding model traces (i.e., model behaviour subset). The accuracy of this approximation depends on the quality of the selected log traces [12]. However, existing log selection techniques (e.g., random, frequency-based [5], K-Medoids [14]) often lack behavioural diversity and model representativeness (see Section II), leading to reduced accuracy in the conformance approximation. Hence, there is significant potential for improving the quality of model behaviour subsets.

In this study, we propose an enhanced model behaviour sampling method to select more representative subsets and obtain more accurate approximate values. First, we apply hierarchical clustering to the event log using our proposed distance criterion. Then, we propose two in-cluster methods to select typical traces from each cluster, which are then used to construct more representative model behaviour subsets. Finally, we extend the existing cost lower bound algorithm to achieve more accurate approximation results. The experimental results show that our approach yields more accurate approximations than existing baselines, though with an increased approximation time.

The remainder of this study is organized as follows. Section II provides a motivating example to further illustrate the research problem. Section III discusses related work in approximate conformance checking. Section IV outlines the necessary preliminaries. In Section V, we propose our method for constructing model behaviour subsets using hierarchical clustering. Section VI details the evaluation setup. Section VII presents the experimental results, followed by Section VIII which presents the discussion. Finally, Section IX concludes the study and presents the limitations and future work.

II. Motivating Example

Research such as [5] and [15] has shown that selecting more typical log traces lead to higher approximation accuracy. Thus, the key challenge is determining which subset should be selected to improve approximate accuracy. Existing log selection methods, such as the frequency-based and K-medoids approaches, sometimes lack sufficient log representativeness.

To illustrate the potential limitations of these methods, we use a synthesized event log L. It contains 5,106 traces consisting of 32,600 events and 12 trace variants, as shown in Table I.

TABLE I. EVENT LOG

ID	Trace Variant	Freq	ID	Trace Variant	Freq
0	$\langle a, b, c, d, f, e, g, h\rangle$	1280	6	$\langle a, d, f, h\rangle$	250
1	$\langle a, b, c, d, e, f, g, h\rangle$	912	7	$\langle a, f, b, c\rangle$	96
2	$\langle a, b, c, d, e, g, f, h\rangle$	864	8	$\langle a, c, e, f, g\rangle$	64
3	$\langle a, b, c, h\rangle$	792	9	$\langle a, d, e, g, h\rangle$	56
4	$\langle a, b, c, d, h\rangle$	400	10	$\langle a, b, f, e, g, h\rangle$	48
5	$\langle a, h\rangle$	320	11	$\langle b, f, g\rangle$	24

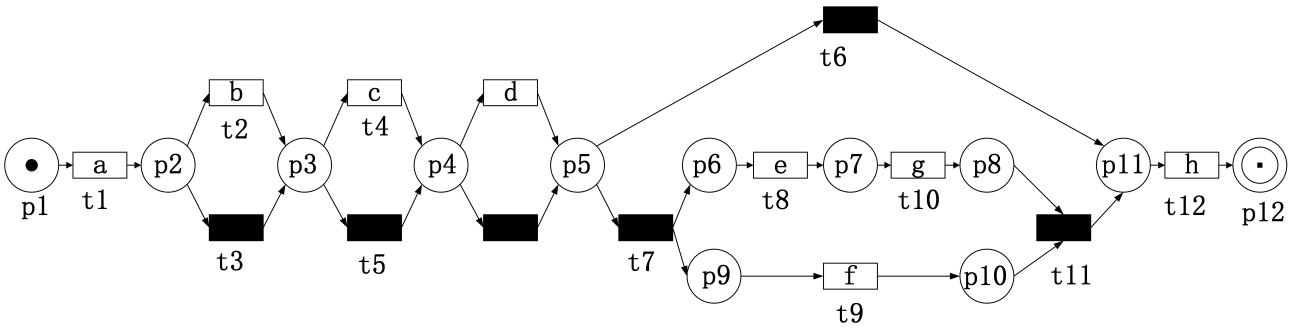


Fig. 1. The process model discovered by inductive miner with infrequent threshold equals to 0.9.

TABLE II. BEHAVIOUR SUBSETS CONSTRUCTED BY FOUR METHODS

Method	Subset	Result	Cost Deviation
Frequency-based	Log Behaviour	$\Sigma_L = \{\langle a, b, c, d, f, e, g, h\rangle, \langle a, b, c, d, e, f, g, h\rangle, \langle a, b, c, d, e, g, f, h\rangle\}$	7806
	Model Behaviour	$\Sigma_M = \{\langle a, b, c, d, f, e, g, h\rangle, \langle a, b, c, d, e, f, g, h\rangle, \langle a, b, c, d, e, g, f, h\rangle\}$	
K-Medoids	Log Behaviour	$\Sigma_L = \{\langle a, h\rangle, \langle a, b, c, d, e, g, f, h\rangle, \langle b, f, g\rangle\}$	6596
	Model Behaviour	$\Sigma_M = \{\langle a, h\rangle, \langle a, b, c, d, e, g, f, h\rangle, \langle a, b, e, f, g, h\rangle\}$	
In-cluster frequency	Log Behaviour	$\Sigma_L = \{\langle a, h\rangle, \langle a, b, c, d, f, e, g, h\rangle, \langle a, b, c, h\rangle\}$	4698
	Model Behaviour	$\Sigma_M = \{\langle a, h\rangle, \langle a, b, c, d, f, e, g, h\rangle \langle a, b, c, h\rangle\}$	
In-cluster medoid	Log Behaviour	$\Sigma_L = \{\langle a, d, f, h\rangle, \langle a, b, c, d, f, e, g, h\rangle, \langle a, b, c, h\rangle\}$	4854
	Model Behaviour	$\Sigma_M = \{\langle a, d, h\rangle, \langle a, b, c, d, f, e, g, h\rangle \langle a, b, c, h\rangle\}$	

To discover the event log presented in Fig. 1, we applied the Inductive Miner algorithm [16] with infrequent thresholds of 0.9.

Assuming that we select three variants to represent the event log, i.e., the behavior subset consists of three variants. Table II shows the behaviour subsets generated by the frequency-based method, K-Medoids, and our proposed methods (see Section V for details). The frequency-based subsets show two key limitations:

1) Overestimation of alignment cost: Variant 5, $\langle a, h\rangle$, can be perfectly replayed in the model with an alignment cost of 0. But it is not included in our model behaviour subset, aligning it would require at least 6 insertions (i.e., cost of 6), resulting in an overestimated approximate cost.

2) Lack of structural diversity: The selected model traces $\langle a, b, c, d, f, e, g, h\rangle$ and $\langle a, b, c, d, e, f, g, h\rangle$ differ only in the order of e and f. This means that they represent essentially the similar structural path, potentially overlooking other important paths in the process model.

Also, the K-Medoids method has drawbacks: it clusters traces solely based on their control-flow information, that is, syntactic difference. For example, the trace $\langle b, f, g\rangle$ in log behaviour subset (as shown in Table II) may have significantly syntactic differences from other traces but, due to its low

frequency (only 24 occurrences), it is still not enough to represent the model behaviour.

To address the issues, our approach proposed in Section V effectively balances frequency and control-flow information. Table II also shows the cost deviation. It refers to the difference in alignment cost between using model behaviour subset and optimal-alignment. The values indicate that the model behaviours generated by our methods significantly reduce the cost deviations compared to vanilla methods.

III. Related Work

To cope with the complexity of alignment construction, approximation techniques have been developed to balance result quality and computational cost. Early studies explored replacing the A* algorithm with faster algorithms[7], [17], [18], laying the foundation for a more efficient alignment computation. For example, Taymouri and Carmona [17], introducing an evolutionary algorithm to enhance alignment approximations. Model decomposition has also been investigated as an efficiency-oriented approximation approach. The foundational work demonstrated how breaking models into smaller and more manageable parts can simplify alignment, although it may not always result in optimal alignments [19], [20]. Furthermore, the construction of automata capable of aligning the log and the model has been explored as another approximation technique [21], [22]. This approach provides good approximations of the optimal alignments in most cases. Recently, some researchers have proposed using RNN-based neural networks to obtain recall and precision metrics for event logs and process models, demonstrating the potential of this technique for conformance analysis [23], [24].

Reducing behaviour size is another promising strategy for approximate conformance checking. One sampling approach focuses on sampling the event log. For instance, [25] proposes a trace sampling method, assuming that a few log traces can estimate the conformance value. However, it lacks upper and lower bounds for the approximation and performs worse when the event log contains many unique behaviors.

Another recent sampling approach targets model behaviour. [5] introduced a model sampling method to construct subsets of the model behaviour that represent the entire process model, significantly reducing the approximation time while largely maintaining accuracy. The method also provides upper and lower bounds to give some certainty of the approximation.

Hierarchical clustering is widely used in process mining for its structural representativeness [26]. Furthermore, [27] demonstrates how hierarchical clustering aids in discovering a better model.

IV. Preliminaries

This section presents the terminology and notation for conformance checking to support the subsequent sections. We use the basic definitions of Petri net, e.g., labeled Petri Net in [28].

Given a system net SN, $\phi_f(SN)$ is the set of all complete firing sequences of SN and $\phi_v(SN)$ is the set of all possible visible traces, i.e., complete firing sequences starting in its initial marking and ending in its final marking projected onto the set of observable activities (not silent transitions, e.g., t_3 in Fig. 1).

To measure how a trace aligns to a process model, moves are represented by pairs (a, t), where a is a log activity, and t is a model transition. Legal moves can be: log moves , model moves, or synchronous moves . Any other combination is an illegal move.

Definition 1. ***(Alignment).*** *Let $\sigma_L \in L$ represent a log trace and $\sigma_M \in \phi_f(SN)$ denote a complete firing sequence of a system net SN. A_{LM} is the set of legal moves. An alignment of σ_L and σ_M is a sequence of pairs $\gamma \in A_{LM}^*$ such that the projection on the first element (ignoring $\gg$) yields σ_L and the projection on the second element (ignoring $\gg$ and transition labels) yields σ_M.*

To quantify the costs of alignments we introduce a cost function δ in Definition 2.

Definition 2. ***(Cost of Alignment).*** *Cost function $\delta \in A_{LM} \to \mathbb{N}$ assigns costs to legal moves. The cost of an alignment $\gamma \in A_{LM}^*$ is the sum of all costs:*

$$\delta(\gamma) = \sum_{(a,t)\in\gamma} \delta(a, t).$$

The cost values assigned to log moves, model moves, and synchronous moves are 1, 1, and 0, respectively. Note that an alignment is considered optimal if it has the minimum alignment cost.

Definition 3. ***(Optimal Alignment).*** *Let L be an event log and SN a system net, where $\phi_v(SN) \neq \emptyset$.*

- *For $\sigma_L \in L$, we define: $\Gamma_{\sigma_L,SN} \in \{\gamma \in A_{LM}^* \mid \exists \sigma_M \in \phi_f(SN)$ is an alignment of σ_L and $\sigma_M\}$.*
- *An alignment $\gamma \in \Gamma_{\sigma_L,SN}$ is optimal for trace $\sigma_L \in L$ and system net SN if for any alignment $\gamma' \in \Gamma_{\sigma_L,M}$: $\delta(\gamma') \geq \delta(\gamma)$.*
- *$\gamma_{SN} \in A_{LM}^* \to A_{LM}^*$ is a mapping that assigns any log trace σ_L to an optimal alignment, i.e., $\gamma_{SN}(\sigma_L) \in \Gamma_{\sigma_L,SN}$ and $\gamma_{SN}(\sigma_L)$ is an optimal alignment.*

Definition 4. ***(Levenshtein Edit Distance).*** *As defined by [29], the Levenshtein edit distance $d(\sigma_1, \sigma_2) \to \mathbb{N}$ represents the minimum number of edit operations (i.e., insertions, deletions, and substitutions) required to transform one sequence into another. For instance, $d(\langle a, b\rangle, \langle c, d\rangle) = 2$, where the two edit operations are substitutions (a, c) and (b, d).*

Definition 5. ***(Edit Distance Cost Function).*** *We can calculate the distance between two traces (or sequences) faster by using a modified version of the Levenshtein edit distance [30]. Let $\sigma_1, \sigma_2 \in A^*$ be two sequences of activities. The Edit Distance Cost Function $\Delta(\sigma_1, \sigma_2) \to \mathbb{N}$ is defined as the minimum number of edits (insertion or deletion of activities) required to transform σ_1 into σ_2.*

Suppose that S is a set of sequences, $\Phi(\sigma_L, S) = \min_{\sigma_M \in S} \Delta(\sigma_L, \sigma_M)$ returns the distance of the most similar sequence in S. Let $\phi_v(SN)$ be the set of all visible firing sequences in SN, and $\gamma_{SN}(\sigma)$ be an optimal alignment for sequence σ. It is possible to prove that $\delta_S(\gamma_{SN}(\sigma)) = \Phi(\sigma, \phi_v(SN))$[12].

In the context of alignment, the edit distance function can be used as a cost function δ_S for evaluating the misalignment between a log trace σ_L and a model trace σ_M. This cost function assigns a value corresponding to the number of operations required to align the two sequences. For example, $\Delta(\langle a, c, b, e, d\rangle, \langle a, b, c, a, d\rangle) = 4$ corresponds to two deletions and two insertions.

Moreover, the alignment cost of a single trace can be converted into a fitness value between 0 (poor fitness, i.e., maximal costs) and 1 (perfect fitness, i.e., zero costs) using Eq. (1) [5]. In this regard, we normalize this cost relative to the worst case, with one log move for each activity in the trace and one model move for each transition in the model's shortest path, $SPM = \min_{\sigma_M \in \phi_f}(|\sigma_M|)$. Here, the optimal alignment cost, $\delta(\gamma_{SN}(\sigma))$, can be replaced by an alternative cost (e.g., edit distance cost) to obtain a corresponding fitness value.

$$Fitness_{\text{Trace}}(\sigma_L, SN) = 1 - \frac{\delta_S(\gamma_{SN}(\sigma))}{|\sigma_L| + SPM} \tag{1}$$

Note that the overall fitness between the event log and the system net is the weighted average of single trace fitness values.

V. Method

In this section, we present the proposed conformance approximation method. An overview of our approach is shown in Fig. 2. The method begins with a preprocessing stage using hierarchical clustering techniques. Next, two methods are proposed for constructing model behaviour subsets: in-cluster frequency and in-cluster medoid methods. Finally, the alignment approximation process is explained.

A. Preprocess Event Log Using Hierarchical Clustering

In this stage, we apply agglomerative hierarchical clustering [31] on event logs. Specifically, we first partition the event log based on trace variants to get the trace variant subset Σ_{σ_v}. Then, we introduce the normalized weighted Levenshtein distance to measure the distance between these variants (see Definition 6) as a new in-cluster distance criterion. This criterion considers both frequency and control-flow information, alleviating the problem with current log selection methods mentioned in Section II. It is used to build a distance matrix, then forming a dendrogram. By cutting-off the dendrogram, we obtain the desired number of clusters. The framework is illustrated in Fig. 3.

Definition 6. ***(Normalized Weighted Levenshtein Distance).*** *Let A^* be the set of all possible sequences of activities in A, and let σ_{v1}, σ_{v2} be two trace variants $\in A^*$. The normalized weighted Levenshtein distance between σ_{v1} and σ_{v2}, where each trace variant has a frequency $f(\sigma_{v1})$ and $f(\sigma_{v2})$, is defined as Eq. (2):*

$$d_{weighted}(\sigma_{v1}, \sigma_{v2}) = \frac{f(\sigma_{v1}) \cdot f(\sigma_{v2}) \cdot d_N(\sigma_{v1}, \sigma_{v2})}{\max\{f(\sigma_{v1})^2, f(\sigma_{v2})^2\}} \tag{2}$$

where, the normalized Levenshtein distance $d_N(\sigma_{v1}, \sigma_{v2})$ is given by Eq. (3):

$$d_N(\sigma_{v1}, \sigma_{v2}) = \frac{d(\sigma_{v1}, \sigma_{v2})}{\max\{|\sigma_{v1}|, |\sigma_{v2}|\}} \tag{3}$$

Here, $d_N(\sigma_{v1}, \sigma_{v2}) = 0$ means the two traces are exactly the same, and $d_N(\sigma_{v1}, \sigma_{v2}) = 1$ means the two traces are completely different.

Definition 7. ***(Distance Matrix).*** *Let $\sigma_{v1}, \sigma_{v2}, \ldots, \sigma_{vi} \in A^*$ represent all trace variants in event log L. The matrix $D(L)$ is defined as Eq. (4):*

$$D(L) = \begin{bmatrix} 0 & d(\sigma_{v1}, \sigma_{v2}) & \cdots & d(\sigma_{v1}, \sigma_{vi}) \\ d(\sigma_{v2}, \sigma_{v1}) & 0 & \cdots & d(\sigma_{v2}, \sigma_{vi}) \\ \vdots & \vdots & \ddots & \vdots \\ d(\sigma_{vi}, \sigma_{v1}) & d(\sigma_{vi}, \sigma_{v2}) & \cdots & 0 \end{bmatrix} \tag{4}$$

where, d is the normalized weighted Levenshtein distance function.

B. Constructing Model Behaviour

In this stage, we first propose two in-cluster methods to get log behaviour subset Σ_L from the generated clusters and transform it into the model behaviour subset Σ_M. Specifically,

1) Candidate selection: After preprocessing, we obtain several clusters, each representing different behaviours within the model. The following question is how to choose the most representative traces from each cluster to construct a more effective log behaviour subset. Existing approaches in approximate conformance checking often rely on either random sampling or frequency-based selection without considering control-flow similarity, which may lead to biased or suboptimal subsets when the frequency distribution is highly imbalanced or when rare but structurally central behaviours exist. To address this, we extend the ideas of frequency-based and medoid selection by introducing two in-cluster methods — the in-cluster frequency method and the in-cluster medoid method — designed to balance efficiency and representativeness.

The in-cluster frequency method selects, from each cluster, the trace variant with the highest frequency of occurrence. This approach assumes that the most common behaviour within a cluster is also the most representative of that cluster's behaviour. Its main advantage lies in computational efficiency, as it does not require computing pairwise distances between traces. Compared to methods that sample traces uniformly at random [25], the frequency method reduces the risk of including low-relevance traces, especially in large-scale logs.

The in-cluster medoid method, in contrast, selects the trace variant that minimises the total Levenshtein distance to all other traces in the cluster, effectively identifying the "central" trace in terms of control-flow similarity. Specifically, it computes the pairwise Levenshtein distances between all traces in each cluster, then constructs a distance matrix and obtains the medoid trace (see Definition 8). This ensures that the selected trace best represents the structural characteristics of its cluster, even if it is not the most frequent. Compared to traditional frequency-only methods, the medoid approach mitigates the bias towards dominant behaviours and is more robust when clusters contain diverse but equally important behaviours.

2) Optimal-alignment: In this step, we align Σ_L with process model to construct the Σ_M, that is, we compute the optimal alignments of selected traces in the event log and finding the corresponding model traces for these alignments.

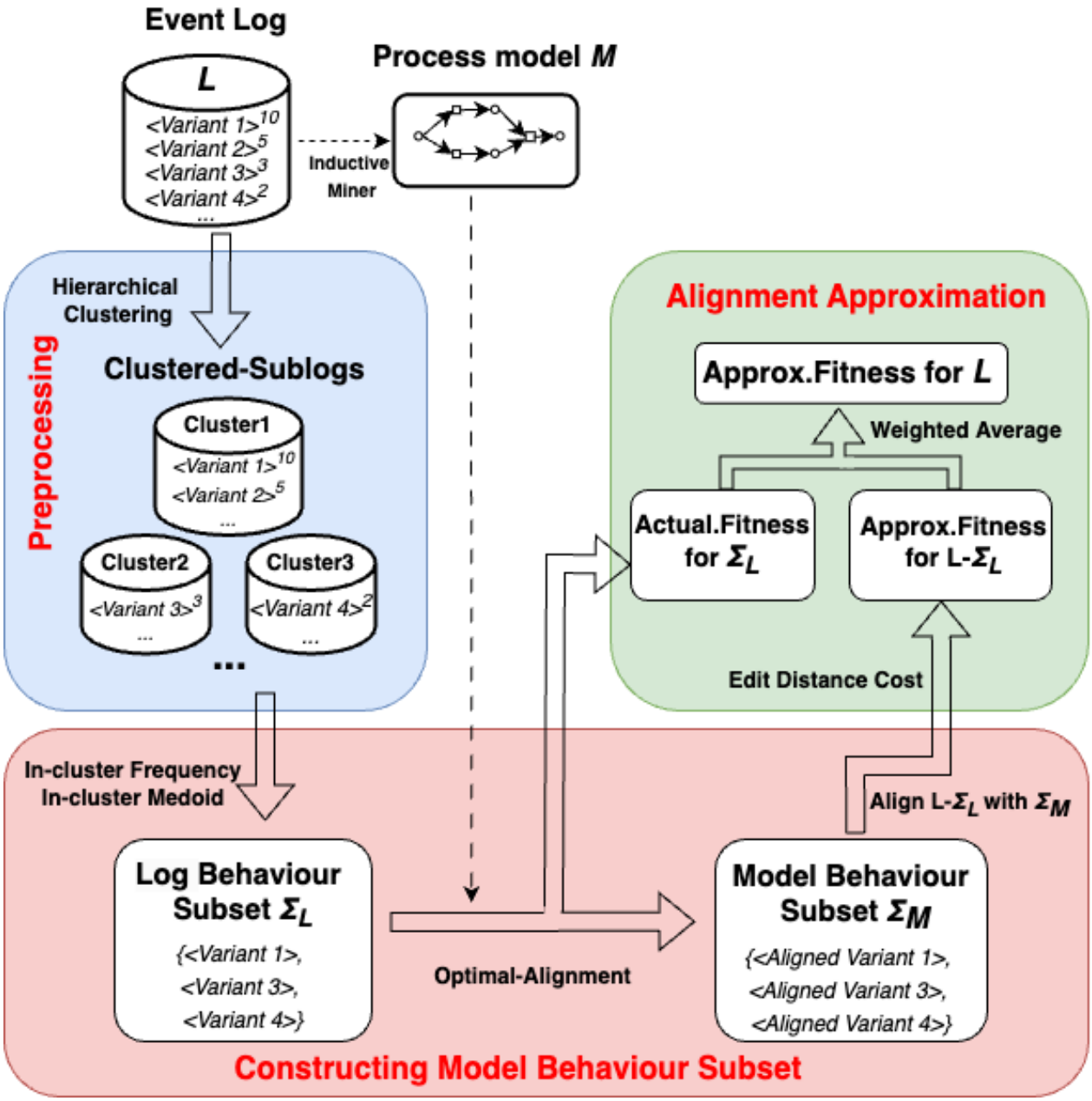


Fig. 2. Overview of our approach.

Table III shows three clusters generated from the event log in Table I. For example, applying the in-cluster frequency method to cluster 2 yields $\langle a, b, c, h\rangle^{792}$, the most frequent trace. Repeating this for each cluster, we obtain $\Sigma_L = \{\langle a, b, c, d, f, e, g, h\rangle^{1280}, \langle a, b, c, h\rangle^{792}, \langle a, h\rangle^{320}\}$. We then align Σ_L with the process model, as shown in Fig. 1, resulting in Σ_M.Note that Σ_L and Σ_M are same in this example, as all traces can be fully replayed in the model.

The specific algorithm steps for proposed methods are outlined in Algorithm 1 and Algorithm 2.

Definition 8. ***(In-cluster Medoid).*** *Let L' be a clustered sublog, n denote the number of trace variants in L', and $D(L')$ be the distance matrix of L'. The trace $\sigma_j = \arg\min_{\sigma_j \in L'} \sum_{i \in [1,n]} d(\sigma_i, \sigma_j)$ represents the medoid trace of sublog L'.*

C. Computing Alignment Approximation

After constructing M_B, we use it to approximate alignments for the traces in $L - L_C$, where L_C refers to the frequency-based trace variants used to build Σ_L. The actual alignment fitness for the variants in Σ_L has already been computed during the construction of M_B, so we can directly use this value for more accurate approximations. At this stage, we calculate the alignment approximations for the remaining variants.

Typically, the actual fitness is calculated using standard alignment costs. However, for the remaining variants, we use

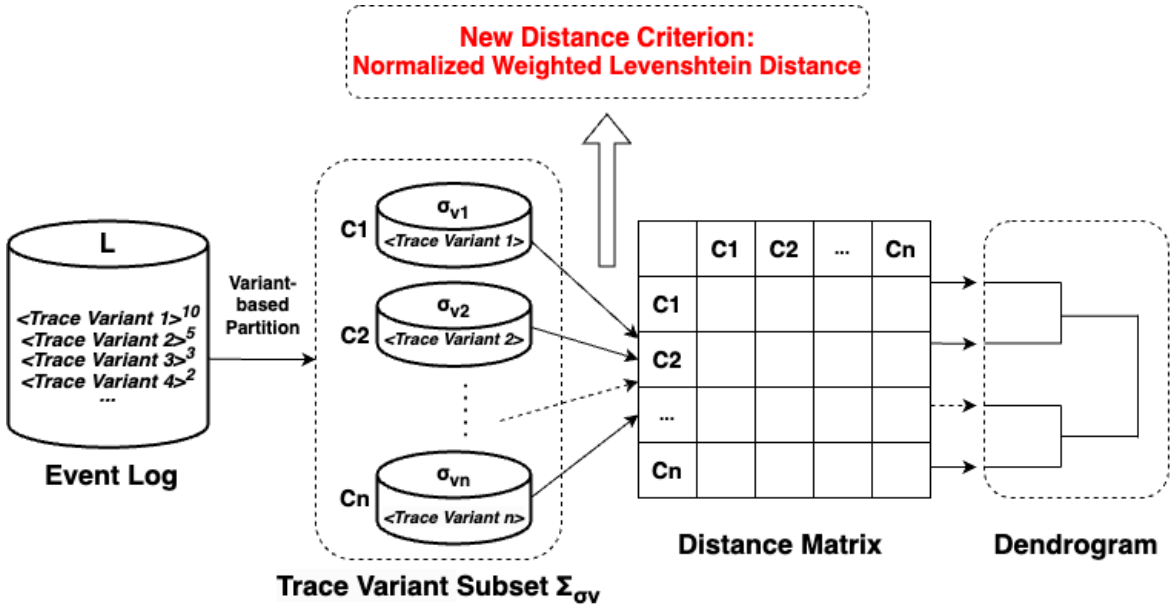


Fig. 3. Preprocessing workflow for hierarchical clustering.

TABLE III. THE CLUSTERS GENERATED FROM THE EXAMPLE LOG PROVIDED IN TABLE I

Cluster ID	Traces in each cluster
1	$\{\langle a,b,c,d,f,e,g,h\rangle^{1280}, \langle a,b,c,d,e,f,g,h\rangle^{912}, \langle a,b,c,d,e,f,g,h\rangle^{864}\}$
2	$\{\langle a,b,c,h\rangle^{792}, \langle a,b,c,d,h\rangle^{400}, \langle a,f,b,c\rangle^{96}\}$
3	$\{\langle a,h\rangle^{320}, \langle a,d,h\rangle^{250}, \langle a,c,e,f,g\rangle^{64}, \langle a,d,e,g,h\rangle^{56}, \langle a,b,f,e,g,h\rangle^{48}, \langle b,f,g\rangle^{24}\}$

Algorithm 1 In-cluster Medoid Method

Input: Event log L; Process model M.
Output: Model behaviour subset Σ_M.

Initialize log behaviour subset: $\Sigma_L \leftarrow \emptyset$
Initialize model behaviour subset: $\Sigma_M \leftarrow \emptyset$
Partition L based on variants into Σ_{σ_v}
Cluster Σ_{σ_v} into k clusters $\{\Sigma_{\sigma_{v1}}, \Sigma_{\sigma_{v2}}, \ldots, \Sigma_{\sigma_{vk}}\}$ using hierarchical clustering
for $i = 1$ to k **do**
    Compute pairwise Levenshtein distances between all variants in $\Sigma_{\sigma_{vi}}$
    Construct distance matrix $D(\Sigma_{\sigma_{vi}})$
    Find the medoid trace $\sigma_L^{(i)}$ in $\Sigma_{\sigma_{vi}}$:

$$\sigma_L^{(i)} = \arg\min_{\sigma \in \Sigma_{\sigma_{vi}}} \sum_{\sigma' \in \Sigma_{\sigma_{vi}}} d(\sigma, \sigma')$$

    Update log behaviour subset: $\Sigma_L \leftarrow \Sigma_L \cup \{\sigma_L^{(i)}\}$
end for
for each $\sigma_L^{(i)} \in \Sigma_L$ **do**
    Compute optimal alignment γ_{SN}^{opt} between $\sigma_L^{(i)}$ and M
    Map to model trace: $\sigma_M^{(i)} \leftarrow \lambda_{SN}(\sigma_L^{(i)})$
    Update model behaviour subset: $\Sigma_M \leftarrow \Sigma_M \cup \{\sigma_M^{(i)}\}$
end for
return Σ_M

Algorithm 2 In-cluster Frequency Method

Input: Event log L; Process model M.
Output: Model behaviour subset Σ_M.

Initialize log behaviour subset: $\Sigma_L \leftarrow \emptyset$
Initialize model behaviour subset: $\Sigma_M \leftarrow \emptyset$
Partition L based on variants into Σ_{σ_v}
Cluster Σ_{σ_v} into k clusters $\{\Sigma_{\sigma_{v1}}, \Sigma_{\sigma_{v2}}, \ldots, \Sigma_{\sigma_{vk}}\}$ using hierarchical clustering
for $i = 1$ to k **do**
    Let $\Sigma_{\sigma_{vi}}$ denote the i-th cluster of variants
    Find the most frequent variant $\sigma_L^{(i)}$ in $\Sigma_{\sigma_{vi}}$:

$$\sigma_L^{(i)} = \arg\max_{\sigma \in \Sigma_{\sigma_{vi}}} f(\sigma)$$

    Update log behaviour subset: $\Sigma_L \leftarrow \Sigma_L \cup \{\sigma_L^{(i)}\}$
end for
for each $\sigma_L^{(i)} \in \Sigma_L$ **do**
    Compute optimal alignment γ_{SN}^{opt} between $\sigma_L^{(i)}$ and M
    Map to model trace: $\sigma_M^{(i)} \leftarrow \lambda_{SN}(\sigma_L^{(i)})$
    Update model behaviour subset: $\Sigma_M \leftarrow \Sigma_M \cup \{\sigma_M^{(i)}\}$
end for
return Σ_M

the edit distance cost function Δ (see Definition 5) to estimate fitness. This method provides guaranteed upper and lower bounds for the alignment cost, instead of exact values [5] (see Lemma 1 and Lemma 2 below) [see Eq. (5)].

$$Fitness(L, SN) = \frac{\sum_{\sigma \in L_C} f(\sigma) \times Fitness_{\text{Approximate}}(\sigma, SN)}{\sum_{\sigma \in L} f(\sigma)} + \frac{\sum_{\sigma \in L - L_C} f(\sigma) \times Fitness_{\text{Actual}}(\sigma, SN)}{\sum_{\sigma \in L} f(\sigma)} \quad (5)$$

Lemma 1 (Alignment Cost Upper Bound)**.** *Let $\sigma_L \in \mathcal{U}_A^*$ be a log trace and $\sigma_M \in \phi_v(SN)$ be a visible firing sequence of SN. We have $\delta_S(\gamma_{SN}(\sigma_L)) \leq \Delta(\sigma_L, \sigma_M)$, where $\gamma_{SN}(\sigma_L)$ is the optimal alignment.*

Proof: The proof is provided in Appendix A and demonstrates how the edit distance guarantees this upper bound. ■

Simply put, if we align trace variant 4 $\langle a, b, c, d, h \rangle$ from Table I with σ_L from the in-cluster frequency subset in Table II, the alignment cost is 1 (i.e., removing "d"). However, since σ_M is a subset of the full model, the actual cost could be smaller or equal. Thus, we use 1 as the upper bound for this variant.

Lemma 2 (Alignment Cost Lower Bound)**.** *Let $SPM = \min_{\sigma_M \in \phi_v(SN)} |\sigma_M|$ and $LPM = \max_{\sigma_M \in \phi_v(SN)} |\sigma_M|$, representing the shortest and longest paths in the process model M. $\sigma_L\lceil_{A_v(SN)}$ and $\kappa(\sigma_L)$ are as defined in Definition 9.*

For any log trace σ_L, if $|\sigma_L\lceil_{A_v(SN)}| < SPM$, the alignment cost lower bound is $SPM - |\sigma_L\lceil_{A_v(SN)}| + \kappa(\sigma_L)$; if $|\sigma_L\lceil_{A_v(SN)}| > LPM$, the lower bound is $|\sigma_L\lceil_{A_v(SN)}| - LPM + \kappa(\sigma_L)$; if $SPM \leq |\sigma_L\lceil_{A_v(SN)}| \leq LPM$, the lower bound is $\kappa(\sigma_L)$.

Proof: The proof is provided in Appendix B. ■

The cost lower bound is the minimum edit operations needed to transform σ_L into σ_M. We refine this algorithm using activity projection (see Definition 9) to improve approximation accuracy. Existing methods compare log trace length directly with the model's range, potentially yielding errors if irrelevant activities are present. For instance, in Fig. 1, a trace $\langle a, x \rangle$ might seem aligned if its length falls within the model's shortest (SPM=2) and longest paths (LPM=8), even though x is not in the model, resulting in a miscalculated cost of 0. Our algorithm removes non-model activities (e.g., removing x from $\langle a, x \rangle$ to form $\langle a \rangle$) before comparing trace lengths. This adjustment yields a more accurate cost of 1 rather than 0, resulting in a smaller upper fitness and tighter bound width.

These bounds are then used to compute the corresponding upper and lower fitness bounds (with the cost upper bound giving the fitness lower bound, and vice versa) using Eq. (1). The computations for the fitness bounds are provided in Algorithm 3 and Algorithm 4. The average of these bounds provides the approximate fitness. Once we compute the approximate fitness for each remaining variant, we take the weighted average of these values along with the previously computed actual fitness to get the overall approximate fitness for the entire event log, as shown in Eq. (5).

Definition 9 (Activity Projection)**.** *Let $A_v(SN)$ be the set of unique observable activities in the system net SN. For any log trace σ_L, let $\sigma_L\lceil_{A_v(SN)}$ represent the projection of σ_L onto $A_v(SN)$, which means the set of activities in σ_L that also appear in the model. Define $\kappa(\sigma_L) = |\sigma_L| - |\sigma_L\lceil_{A_v(SN)}|$ as the number of activities in σ_L that are not present in the model.*

For example, let $\sigma_L = \langle a, b, x \rangle$ be a log trace and the observable activities of the system net be $A_v(SN) = \{a, b, c, d, e\}$. Projecting σ_L onto $A_v(SN)$ results in $\sigma_L\lceil_{A_v(SN)} = \langle a, b \rangle$, as x is not part of $A_v(SN)$. Therefore, $\kappa(\sigma_L) = |\sigma_L| - |\sigma_L\lceil_{A_v(SN)}| = 3 - 2 = 1$, indicating one activity in σ_L is not present in the model.

Algorithm 3 Fitness lower bound computation

Input: Event log L; Optimal-aligned Log L_C; Model behaviour subset Σ_M.
Output: Lower bound fitness $L_fitness(\sigma_L, M)$.

for each $\sigma_L \in L - L_C$ **do**
  $\Phi(\sigma_L, \Sigma_M)$ // Compute minimun edit distance cost
  $L_fitness(\sigma_L, M) \leftarrow 1 - \frac{\Phi(\sigma_L, \Sigma_M)}{|\sigma_L| + \min_{\sigma_M \in \phi_v(SN)}(|\sigma_M|)}$
end for
return $L_fitness(\sigma_L, M)$

Algorithm 4 Fitness upper bound computation

Input: Event log L; Optimal-aligned Log L_C; Model behaviour subset Σ_M.
Output: Upper bound fitness $U_fitness(\sigma_L, M)$.

$SPM \leftarrow \min_{\sigma_M \in \phi_v(SN)} |\sigma_M|$ // Shortest path
$LPM \leftarrow \max_{\sigma_M \in \phi_v(SN)} |\sigma_M|$ // Longest path
for each $\sigma_L \in L - L_C$ **do**
  Project σ_L onto SN: $\sigma_L\lceil_{A_v(SN)}$
  Compute $\kappa(\sigma_L) = |\sigma_L| - |\sigma_L\lceil_{A_v(SN)}|$
  if $|\sigma_L\lceil_{A_v(SN)}| < SPM$ **then**
    $U_fitness(\sigma_L, M) \leftarrow 1 - \frac{SPM - |\sigma_L\lceil_{A_v(SN)}| + \kappa(\sigma_L)}{|\sigma_L| + \min_{\sigma_M \in \phi_v(SN)}(|\sigma_M|)}$
  else if $|\sigma_L\lceil_{A_v(SN)}| > LPM$ **then**
    $U_fitness(\sigma_L, M) \leftarrow 1 - \frac{|\sigma_L\lceil_{A_v(SN)}| - LPM + \kappa(\sigma_L)}{|\sigma_L| + \min_{\sigma_M \in \phi_v(SN)}(|\sigma_M|)}$
  else
    $U_fitness(\sigma_L, M) \leftarrow 1 - \frac{\kappa(\sigma_L)}{|\sigma_L| + \min_{\sigma_M \in \phi_v(SN)}(|\sigma_M|)}$
  end if
end for
return $U_fitness(\sigma_L, M)$

VI. Evaluation

In this section, we assess the accuracy and time performance of our proposed log selection methods compared to frequency-based and K-Medoids techniques, and evaluate their differences in accuracy and time against normal alignment. Note that the comparison between model behaviour sampling and other approximate methods has been discussed in [5], we focus here on comparisons with the baselines of model behaviour sampling. First, we briefly describe the implementation (Section VI-A) and experimental setup (Section VI-B), followed by a discussion of the experimental results (Section VII).

A. Implementation

Our implementation consists of two steps: first, we implemented the algorithms described in Sections V-A and V-B in Python, to generate log behaviour subsets from event logs.

TABLE IV. THE REAL-LIFE EVENT LOGS USED IN THE EXPERIMENTS

Event Log	Activities #	Traces #	Variants #	Uniqueness
BPIC2012 [36]	25	13087	4366	0.33
BPIC2013-closed problems [37]	4	1487	183	0.12
BPIC2016-Questions [38]	8	21533	2261	0.10
BPIC2017 [39]	28	31509	15930	0.51
Spesis [40]	18	1050	846	0.81
RTFMP [41]	13	150370	231	0.01

Specifically, we extended the pm4py.algo.clustering package in PM4py [32] by introducing the normalized weighted Levenshtein distance (Definition 6), to perform hierarchical clustering. And implemented two proposed in-cluster methods to get the log behaviour subset based on the clustering result. In the second step, we used an existing plugin in ProM [33], *Conformance Log to Log Approximation* [34], with the generated model behaviour subset and the original event log as input, obtaining approximate fitness bounds and values. For the baselines, we used the implementation proposed by Fanisani [5]. For normal alignment, we used PM4py to compute the time and fitness values. The source code and experimental results is available on Github [1].

B. Experimental Setup

Our experiments were based on six real event logs, with basic information about these event logs given in Table IV. Here, *Uniqueness* refers to $\frac{\text{Variant\#}}{\text{Trace\#}}$. A *Uniqueness* value close to 1 indicates that almost all traces are different, e.g., *Sepsis*. For process discovery, we used *Inductive Miner infrequent algorithm* [35] with *infrequent thresholds of 0.4* to get the process model. Two log selection methods, *frequency-based sampling*, *K-Medoids clustering*, were used as baselines to compare with our proposed methods, i.e., *In-cluster frequency method* and *In-cluster medoid method*. Furthermore, we set the *selection percentage* to 10%, 20%, 30%, 40%, and 50%, representing the ratio of the selected variants to the total number of variants in the event logs. Our experiment was repeated four times since the conformance approximation time is non-deterministic. Finally, we performed the experiments on a computer with Apple M1 (8 cores), 8 GB RAM running macOS.

1) Evaluation metrics: To measure approximation accuracy, we used *Approximate Error*, defined as $ApproximateError = |ActualFitness - ApproximateFitness|$, where a value closer to 0 indicates higher accuracy. Additionally, we assess the *Bound Width* as $BoundWidth = U_fitness - L_fitness$, with a smaller width indicating tighter bounds and a more accurate approximation.

We used the *Performance Improvement* (PI) metric, defined as $PI = \frac{\textit{Actual Conformance Time}}{\textit{Approximate Conformance Time}}$ to assess time performance. *Actual Conformance Time* refers to the time needed to compute normal alignment, while *Approximate Conformance Time* includes the total time for the approximation. A PI value greater than 1 indicates the approximation is faster than the actual conformance computation. Preprocessing time (e.g., hierarchical clustering) is included in the approximate conformance time.

[1]https://github.com/lvyl9909/Approximate-Conformance-Checking-using-Hierarchical-Clustering.git

VII. RESULT

Table V presents the *Actual Fitness* and *Approximate Fitness*, *Approximate Error*, and *PI* for four selection methods using 20% of the variants in six event logs. For each metric in a given row, the best value is highlighted in bold. The results show that the proposed in-cluster methods achieve the highest *fitness* and the lowest *approximate error* in most cases, indicating superior accuracy compared to the baselines. In terms of *PI*, the frequency-based method consistently achieves the highest values, reflecting its shorter approximate time. Our complete experimental data is provided in Appendix B.

Fig. 4 shows that both *Approximate Error* and *Bound Width* decrease as the selection percentages increase. Here, *Bound Width* is represented by bars, and *Approximate Error* by lines, illustrating the improvements in these metrics as the selection percentage increases. Our in-cluster methods consistently achieve tighter bounds at each selection percentage. Notably, at a selection 50% in the BPIC2017 log, the bound widths of the baseline are around 0.05, while our methods reduce this by 40% to 0.03. Furthermore, in all data sets with different selection percentages, the in-cluster frequency method shows an average improvement of 19.1% in *Approximate Error* compared to the frequency-based method, while the in-cluster medoid method achieves an average improvement of 27.6% compared to the K-Medoid method. Moreover, the in-cluster frequency method often produces tighter bounds than in-cluster medoid method, especially on low *uniqueness* logs like BPIC2016-Questions, where selecting the most frequent trace is more effective than clustering. However, on high *Uniqueness* logs like *Sepsis*, in-cluster medoid method provides more accurate approximations. In Fig. 5, we compare the time performance of different log selection methods and their improvement over normal alignment. Note that a value of 1 represents the normal alignment time. Consistent with the results in Table V, the frequency method usually yields the highest performance improvement, followed by the K-Medoids method. Our methods are less efficient compared to these baselines, particularly on datasets with higher *Uniqueness* values. Table VI presents the experimental results for datasets.

VIII. DISCUSSION

Across Table V and Fig. 4, our in-cluster methods consistently achieve higher *fitness*, lower *approximate error*, and tighter bounds than the baselines, with the in-cluster frequency method performing better on low-*Uniqueness* logs

TABLE V. APPROXIMATE RESULT COMPARISONS (20% SELECTION) FOR FOUR DIFFERENT SELECTION METHODS

Event Log	Actual	Frequency			K-Medoids			In-cluster freq.			In-cluster medoid		
		Fit.	Err.	PI	Fit.	Err.	PI	Fit.	Err.	PI	Fit.	Err.	PI
BPIC2012	0.9995	0.9741	0.0254	**61.8496**	0.9761	0.0234	41.1727	0.9788	0.0207	25.6113	**0.9806**	**0.0189**	24.8483
BPIC2013-closed problems	0.9997	0.9860	0.0138	**11.8502**	0.9711	0.0286	5.8732	**0.9894**	**0.0103**	1.6728	0.9875	0.0122	1.6443
BPIC2016-Questions	0.9997	0.9923	0.0074	**45.3310**	0.9463	0.0535	30.4731	**0.9944**	**0.0053**	13.1973	0.9565	0.0432	12.2026
BPIC2017	0.9995	0.9690	0.0305	**11.8531**	0.9700	0.0296	9.7231	**0.9749**	**0.0246**	1.9688	0.9747	0.0248	1.8838
Road	0.9999	0.9997	0.0002	**15.7220**	0.9996	0.0004	11.7262	**0.9998**	**0.0001**	7.5686	0.9995	0.0004	6.7700
Sepsis	0.9880	0.9202	0.0679	**53.4338**	0.9202	0.0678	44.9919	0.9313	0.0567	22.9238	**0.9319**	**0.0561**	20.0751

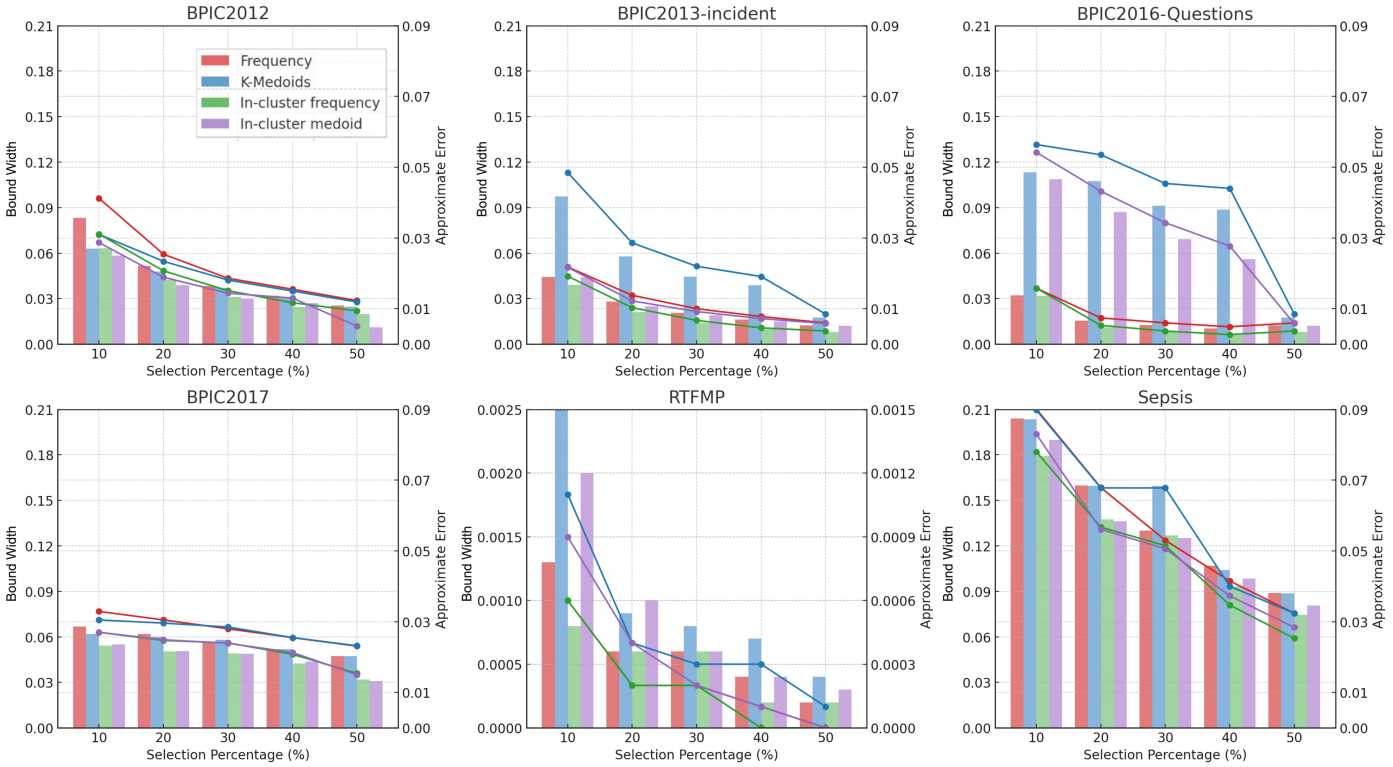


Fig. 4. The performance differences of different selection strategies on band width and approximate error.

(e.g., BPIC2016-Questions) and the in-cluster medoid method excelling on high-*Uniqueness* logs such as *Sepsis*, highlighting a key advantage of our approach over the baselines—improved approximation accuracy. Fig. 5 shows that our methods have larger approximation times. This is because hierarchical clustering requires step-by-step merging and evaluating all possible cluster combinations, which increases preprocessing time compared to baselines. Nevertheless, they remain significantly faster than the normal alignment-based approach, keeping approximation times within acceptable limits while delivering higher accuracy—making them well-suited for large-scale processes where neither a quick, coarse estimate nor weeks of exact computation is desirable. Overall, our results indicate a clear trade-off: the proposed methods bring the approximations closer to the actual values at the cost of some additional but acceptable preprocessing time.

IX. CONCLUSION

In this study, we propose an enhanced model behaviour sampling method using hierarchical clustering to construct more representative model behaviour subsets. By incorporating both frequency and control-flow information from the event log, our approach more effectively captures the model's behaviour, leading to improved approximation accuracy. Experimental results show that our method produces approximations that are on average over 19.1% closer to the actual alignment values than baseline methods, though it requires more computation time.

A potential limitation of this study is the lack of an explicit quantification of how much "increased" time would be acceptable for the "improvement" in accuracy, which is important to evaluate the practical utility of the method under different application scenarios. As a next step, we plan to conduct a systematic, quantitative analysis of the accuracy–time trade-off. Based on it, an incremental approximation tool

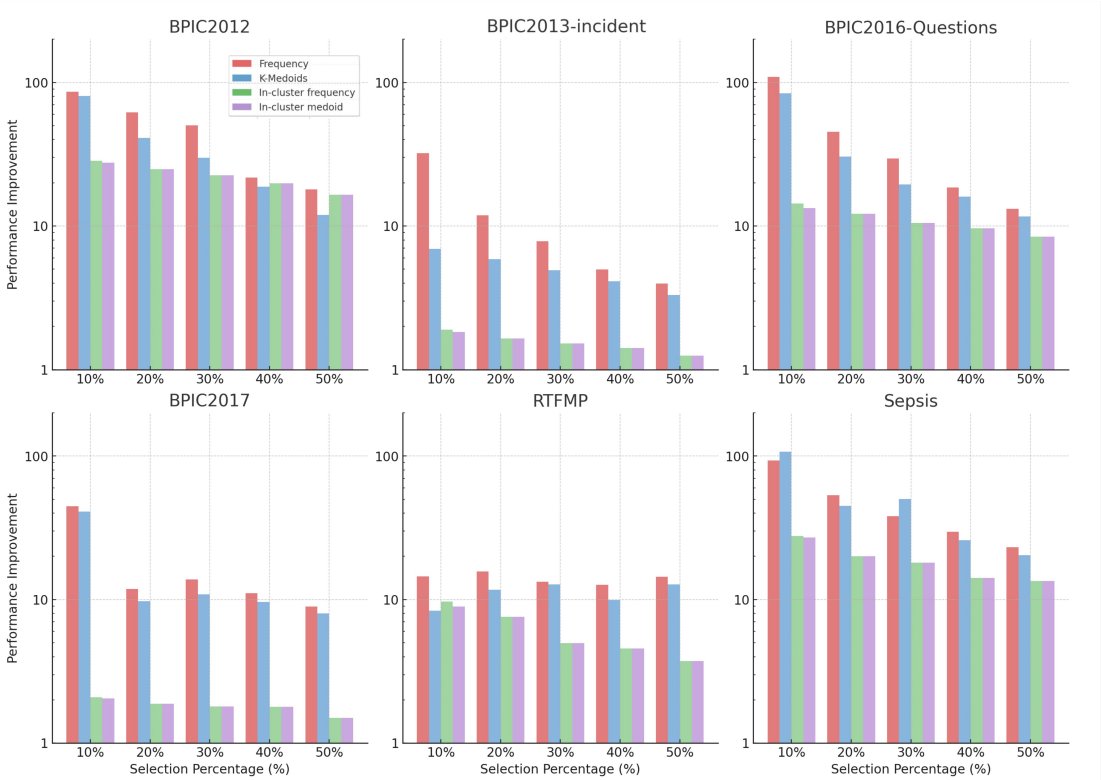


Fig. 5. The performance improvement using different methods in six event logs.

could be developed to increase the size of model behaviour during the time, allowing the user to decide when the accuracy is enough. In addition, we plan to apply a time-optimized hierarchical clustering algorithm to reduce the approximation time of our method. Furthermore, exploring how to make use of the distribution information (e.g., *Uniqueness*) in the event log to choose a better approximate method is also a direction for future research.

ACKNOWLEDGMENT

The authors would like to thank Artem Polyvyanyy for his invaluable guidance and support throughout this research project.

DECLARATION OF COMPETING INTEREST

The author declares that there are no known financial interests or personal relationships that could have influenced the research presented in this study.

REFERENCES

[1] A. A. Mitsyuk, I. A. Lomazova, I. S. Shugurov, and W. M. van der Aalst, "Process model repair by detecting unfitting fragments?" in *Proceedings of the 6th International Conference on Analysis of Images, Social Networks and Texts (AIST 2017)*. CEUR-WS. org, January 2017, pp. 301–313.

[2] W. Van der Aalst, A. Adriansyah, and B. Van Dongen, "Replaying history on process models for conformance checking and performance analysis," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 182–192, 2012.

[3] J. Carmona, B. Dongen, A. Solti, and M. Weidlich, "Conformance checking: Relating processes and models, january 2018."

[4] M. De Leoni and W. M. Van Der Aalst, "Aligning event logs and process models for multi-perspective conformance checking: An approach based on integer linear programming," in *Business Process Management: 11th International Conference, BPM 2013, Beijing, China, August 26-30, 2013. Proceedings*. Springer, 2013, pp. 113–129.

[5] M. Fani Sani, S. van Zelst, and W. van der Aalst, "Conformance checking approximation using subset selection and edit distance," in *In: Dustdar, S., Yu, E., Salinesi, C., Rieu, D., Pant, V. (eds) CAiSE 2020. LNCS*, vol. 12127. Springer, 2020, pp. 234–251.

[6] A. Syamsiyah and B. F. van Dongen, "Improving alignment computation using model-based preprocessing," in *2019 International Conference on Process Mining (ICPM)*. IEEE, 2019, pp. 73–80.

[7] F. Taymouri and J. Carmona, "A recursive paradigm for aligning observed behavior of large structured process models," in *International Conference on Business Process Management*. Springer, 2016, pp. 197–214.

[8] J. Munoz-Gama, J. Carmona, and W. M. Van Der Aalst, "Single-entry single-exit decomposed conformance checking," *Information Systems*, vol. 46, pp. 102–122, 2014.

[9] A. Syamsiyah and B. F. van Dongen, "Improving alignment computation using model-based preprocessing," in *2019 International Conference on Process Mining (ICPM)*. IEEE, 2019, pp. 73–80.

[10] M. Bauer, H. Van der Aa, and M. Weidlich, "Estimating process conformance by trace sampling and result approximation," in *Business Process Management: 17th International Conference, BPM 2019, Vienna, Austria, September 1–6, 2019, Proceedings 17*. Springer, 2019, pp. 179–197.

[11] M. Bauer, H. van der Aa, and M. Weidlich, "Sampling and approximation techniques for efficient process conformance checking," *Information Systems*, vol. 104, p. 101666, 2022.

[12] M. Sani, M. Kabierski, S. Van Zelst *et al.*, "Model independent error bound estimation for conformance checking approximation," Preprint on arXiv, 2021, available at https://arxiv.org/abs/2103.13315.

[13] M. Fani Sani, J. Garza Gonzalez, S. van Zelst, and W. van der Aalst, "Conformance checking approximation using simulation," in *2nd International Conference on Process Mining, ICPM 2020*. Padua, Italy: IEEE, 2020, pp. 105–112.

[14] M. Fani Sani, M. Boltenhagen, and W. van der Aalst, "Prototype selection using clustering and conformance metrics for process discovery," in *Business Process Management Workshops: BPM 2020 International Workshops, Seville, Spain, September 13–18, 2020, Revised Selected Papers 18*. Springer, 2020, pp. 281–294.

[15] ——, "Prototype selection using clustering and conformance metrics for process discovery," in *Business Process Management Workshops: BPM 2020 International Workshops, Seville, Spain, September 13–18, 2020, Revised Selected Papers 18*. Springer, 2020, pp. 281–294.

[16] S. J. Leemans, D. Fahland, and W. M. Van Der Aalst, "Discovering block-structured process models from event logs containing infrequent behaviour," in *Business Process Management Workshops: BPM 2013 International Workshops, Beijing, China, August 26, 2013, Revised Papers 11*. Springer, 2014, pp. 66–78.

[17] F. Taymouri and J. Carmona, "An evolutionary technique to approximate multiple optimal alignments," in *Business Process Management: 16th International Conference, BPM 2018, Sydney, NSW, Australia, September 9–14, 2018, Proceedings 16*. Springer International Publishing, 2018, pp. 215–232.

[18] ——, "Computing alignments of well-formed process models using local search," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 29, no. 3, pp. 1–41, 2020.

[19] J. Munoz-Gama, J. Carmona, and W. M. P. Van Der Aalst, "Single-entry single-exit decomposed conformance checking," *Information Systems*, vol. 46, pp. 102–122, 2014.

[20] W. M. P. Van der Aalst, "Decomposing petri nets for process mining: A generic approach," *Distributed and Parallel Databases*, vol. 31, pp. 471–507, 2013.

[21] S. J. Leemans, D. Fahland, and W. M. Van der Aalst, "Scalable process discovery and conformance checking," *Software & Systems Modeling*, vol. 17, pp. 599–631, 2018.

[22] D. Reißner, A. Armas-Cervantes, R. Conforti, M. Dumas, D. Fahland, and M. La Rosa, "Scalable alignment of process models and event logs: An approach based on automata and s-components," *Information Systems*, vol. 94, p. 101561, 2020.

[23] Y. Lyu, "A mamba-based approximate conformance checking method," in *Fourth International Conference on Advanced Algorithms and Neural Networks (AANN 2024)*, vol. 13416. SPIE, 2024, pp. 885–891.

[24] J. Peeperkorn, S. vanden Broucke, and J. De Weerdt, "Supervised conformance checking using recurrent neural network classifiers," in *International Conference on Process Mining*. Springer, 2020, pp. 175–187.

[25] M. Bauer, H. van der Aa, and M. Weidlich, "Sampling and approximation techniques for efficient process conformance checking," *Information Systems*, vol. 104, p. 101666, 2022.

[26] J.-Y. Jung, J. Bae, and L. Liu, "Hierarchical clustering of business process models," *International Journal of Innovative Computing, Information and Control*, vol. 5, no. 12, pp. 1349–4198, 2009.

[27] S. van Zelst and Y. Cao, "A generic framework for attribute-driven hierarchical trace clustering," in *Business Process Management Workshops. BPM 2020*, ser. Lecture Notes in Business Information Processing, A. Del Río Ortega, H. Leopold, and F. Santoro, Eds., vol. 397. Springer, Cham, 2020.

[28] W. M. P. Van der Aalst, "Decomposing petri nets for process mining: A generic approach," *Distributed and Parallel Databases*, vol. 31, no. 4, pp. 471–507, 2013.

[29] V. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *Proceedings of the Soviet physics doklady*, 1966.

[30] P. H. Sellers, "On the theory and computation of evolutionary distances," *SIAM Journal on Applied Mathematics*, vol. 26, no. 4, pp. 787–793, 1974.

[31] D. Müllner, "Modern hierarchical, agglomerative clustering algorithms," *arXiv preprint arXiv:1109.2378*, 2011.

[32] A. Berti, S. van Zelst, and D. Schuster, "Pm4py: A process mining library for python," *Software Impacts*, vol. 17, p. 100556, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2665963823000933

[33] H. Verbeek, J. Buijs, B. Van Dongen, and W. M. van der Aalst, "Prom 6: The process mining toolkit," *Proc. of BPM Demonstration Track*, vol. 615, pp. 34–39, 2010.

[34] M. F. Sani, J. J. G. Gonzalez, S. J. van Zelst, and W. M. van der Aalst, "Alignment approximator: A prom plug-in to approximate conformance statistics." in *BPM (Demos/Resources Forum)*, 2023, pp. 102–106.

[35] S. J. Leemans, D. Fahland, and W. M. Van Der Aalst, "Discovering block-structured process models from event logs containing infrequent behaviour," in *Business Process Management Workshops: BPM 2013 International Workshops, Beijing, China, August 26, 2013, Revised Papers 11*. Springer, 2014, pp. 66–78.

[36] B. van Dongen, "Bpi challenge 2012," 2012. [Online]. Available: https://data.4tu.nl/articles/dataset/BPI_Challenge_2012/12689204/1

[37] W. Steeman, "Bpi challenge 2013, closed problems," 2013. [Online]. Available: https://data.4tu.nl/articles/dataset/BPI_Challenge_2013_closed_problems/12714476/1

[38] M. Dees and B. van Dongen, "Bpi challenge 2016: Questions," 2016. [Online]. Available: https://data.4tu.nl/articles/dataset/BPI_Challenge_2016_Questions/12687320/1

[39] B. van Dongen, "Bpi challenge 2017," 2017. [Online]. Available: https://data.4tu.nl/articles/dataset/BPI_Challenge_2017/12696884/1

[40] F. Mannhardt, "Sepsis cases - event log," 2016. [Online]. Available: https://data.4tu.nl/articles/dataset/Sepsis_Cases_-_Event_Log/12707639/1

[41] M. M. de Leoni and F. Mannhardt, "Road traffic fine management process," 2015. [Online]. Available: https://data.4tu.nl/articles/dataset/Road_Traffic_Fine_Management_Process/12683249/1

APPENDIX

A. Proof of Alignment Cost Upper Bound

Proof: We have shown that $\min_{\sigma_M \in S} \Delta(\sigma_L, \sigma_M) = \delta_S(\gamma_{SN}(\sigma_L))$ in Definition 5, so $\Delta(\sigma_L, \sigma_M) \geq \delta_S(\gamma_{SN}(\sigma_L))$. Therefore, if $\delta_S(\gamma_{SN}(\sigma_L)) > \Delta(\sigma_L, \sigma_M)$, $\gamma_{SN}(\sigma_L)$ is not an optimal alignment. Consequently, for any $M_B \subseteq \phi_v(SN)$, $\Phi(\sigma_L, M_B)$ returns an upper bound for the cost of optimal alignment [5]. ■

B. Proof of Alignment Cost Lower Bound

Proof: When $|\sigma_L\lceil_{A_v(SN)}| < SPM$, at least $SPM - |\sigma_L\lceil_{A_v(SN)}|$ insertions are needed. Adding the initial alignment cost, the total minimum alignment cost is $|SPM - \sigma_L\lceil_{A_v(SN)}| + |\kappa(\sigma_L)|$. Similarly, when $|\sigma_L\lceil_{A_v(SN)}| > LPM$, at least $|\sigma_L\lceil_{A_v(SN)}| - LPM$ deletions are required. Thus, the total alignment cost is $|\sigma_L\lceil_{A_v(SN)} - LPM| + |\kappa(\sigma_L)|$. When $SPM \leq |\sigma_L\lceil_{A_v(SN)}| \leq LPM$, no insertions or deletions are needed, so the alignment cost is $|\kappa(\sigma_L)|$. ■

TABLE VI. EXPERIMENTAL RESULTS FOR DATASETS.

Log	Actual Fitness	Normal Alignment Time	Candidate Percentage	Parameter		Approximation Method			
						Baseline		In-cluster medoid	
						Frequency	K-Medoids	In-cluster frequency	In-cluster medoid
BPIC2012	0.9995	35400000	10%	Approximate fitness	Lower Bound	0.9167	0.9371	0.9368	0.9416
					Approximate fitness	0.9583	0.9685	0.9684	0.9708
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0412	0.0310	0.0311	0.0287
				Band Width		0.0833	0.0629	0.0632	0.0584
				Preprocessing Time (ms)		/	/	1219923	1259201
				Approximate Time (ms)		411778	439928	25030	26102
				Total Approximate Time (ms)		411778	439928	1244953	1285303
				PI		85.9687	80.4677	28.4348	27.5421
			20%	Approximate fitness	Lower Bound	0.9482	0.9522	0.9576	0.9612
					Approximate fitness	0.9741	0.9761	0.9788	0.9806
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0254	0.0234	0.0207	0.0189
				Band Width		0.0518	0.0478	0.0424	0.0388
				Preprocessing Time (ms)		/	/	1342972	1392321
				Approximate Time (ms)		572356	859792	39232	32323
				Total Approximate Time (ms)		572356	859792	1382204	1424644
				PI		61.8496	41.1727	25.6113	24.8483
			30%	Approximate fitness	Lower Bound	0.9618	0.9629	0.9688	0.9702
					Approximate fitness	0.9809	0.9814	0.9844	0.9851
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0186	0.0181	0.0151	0.0144
				Band Width		0.0382	0.0371	0.0312	0.0298
				Preprocessing Time (ms)		/	/	1423219	1529312
				Approximate Time (ms)		702244	1186892	41992	42223
				Total Approximate Time (ms)		702244	1186892	1465211	1571535
				PI		50.4098	29.8258	24.1603	22.5257
			40%	Approximate fitness	Lower Bound	0.9681	0.9690	0.9756	0.9730
					Approximate fitness	0.9841	0.9845	0.9878	0.9865
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0155	0.0150	0.0117	0.0130
				Band Width		0.0319	0.0310	0.0244	0.0270
				Preprocessing Time (ms)		/	/	1591211	1730030
				Approximate Time (ms)		1229401	1480757	41503	49020
				Total Approximate Time (ms)		1229401	1480757	1632714	1779050
				PI		28.7945	23.9067	21.6817	19.8983
			50%	Approximate fitness	Lower Bound	0.9745	0.9752	0.9802	0.9888
					Approximate fitness	0.9873	0.9876	0.9901	0.9944
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0123	0.0119	0.0094	0.0051
				Band Width		0.0255	0.0248	0.0198	0.0112
				Preprocessing Time (ms)		/	/	1823900	2102097
				Approximate Time (ms)		1863573	1971131	42826	43503
				Total Approximate Time (ms)		1863573	1971131	1866726	2145600
				PI		18.9958	17.9592	18.9637	16.4989

TABLE VI. EXPERIMENTAL RESULTS FOR DATASETS.

Table VI continued.

Log	Actual Fitness	Normal Alignment Time	Candidate Percentage	Parameter		Approximation Method			
						Baseline		In-cluster medoid	
						Frequency	K-Medoids	In-cluster frequency	In-cluster medoid
BPIC2013-incident	0.9997	135400	10%	Approximate fitness	Lower Bound	0.9559	0.9025	0.9610	0.9560
					Approximate fitness	0.9780	0.9513	0.9805	0.9780
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0218	0.0485	0.0192	0.0217
				Band Width		0.0441	0.0975	0.0390	0.0440
				Preprocessing Time (ms)		/	/	69233	70923
				Approximate Time (ms)		4200	19572	2033	2992
				Total Approximate Time (ms)		4200	19572	71266	73915
				PI		32.2381	6.9180	1.8999	1.8318
			20%	Approximate fitness	Lower Bound	0.9719	0.9422	0.9788	0.9750
					Approximate fitness	0.9860	0.9711	0.9894	0.9875
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0138	0.0286	0.0103	0.0122
				Band Width		0.0281	0.0578	0.0212	0.0250
				Preprocessing Time (ms)		/	/	78012	79232
				Approximate Time (ms)		11426	23054	2932	3111
				Total Approximate Time (ms)		11426	23054	80944	82343
				PI		11.8502	5.8732	1.6728	1.6443
			30%	Approximate fitness	Lower Bound	0.9795	0.9554	0.9860	0.9810
					Approximate fitness	0.9898	0.9777	0.9930	0.9905
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0100	0.0220	0.0067	0.0092
				Band Width		0.0205	0.0446	0.0140	0.0190
				Preprocessing Time (ms)		/	/	81203	85003
				Approximate Time (ms)		17294	27553	3504	4092
				Total Approximate Time (ms)		17294	27553	84707	89095
				PI		7.8293	4.9142	1.5985	1.5197
			40%	Approximate fitness	Lower Bound	0.9839	0.9612	0.9902	0.9850
					Approximate fitness	0.9920	0.9806	0.9951	0.9925
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0078	0.0191	0.0046	0.0072
				Band Width		0.0161	0.0388	0.0098	0.0150
				Preprocessing Time (ms)		/	/	89129	91892
				Approximate Time (ms)		27133	32868	3932	3902
				Total Approximate Time (ms)		27133	32868	93061	95794
				PI		4.9902	4.1195	1.4550	1.4134
			50%	Approximate fitness	Lower Bound	0.9875	0.9825	0.9920	0.9879
					Approximate fitness	0.9938	0.9913	0.9960	0.9940
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0060	0.0085	0.0037	0.0058
				Band Width		0.0125	0.0175	0.0080	0.0121
				Preprocessing Time (ms)		/	/	95002	104023
				Approximate Time (ms)		34006	41028	4002	4350
				Total Approximate Time (ms)		34006	41028	99004	108373
				PI		3.9817	3.3002	1.3676	1.2494

TABLE VI. EXPERIMENTAL RESULTS FOR DATASETS.

Table VI continued.

Log	Actual Fitness	Normal Alignment Time	Candidate Percentage	Parameter		Approximation Method			
						Baseline		In-cluster medoid	
						Frequency	K-Medoids	In-cluster frequency	In-cluster medoid
BPIC2016-Questions	0.9997	5200690	10%	Approximate fitness	Lower Bound	0.9679	0.8867	0.9680	0.8911
					Approximate fitness	0.9840	0.9434	0.9840	0.9455
					Upper Bound	1.0000	1.0000	0.9999	0.9999
				Approximation Error		0.0158	0.0564	0.0158	0.0542
				Band Width		0.0321	0.1133	0.0319	0.1088
				Preprocessing Time(ms)		/	/	359923	389454
				Approximate Time(ms)		47607	61807	2715	1551
				Total Approximate Time(ms)		47607	61807	362638	391005
				PI		109.2421	84.1440	14.3413	13.3008
			20%	Approximate fitness	Lower Bound	0.9845	0.8925	0.9888	0.9130
					Approximate fitness	0.9923	0.9463	0.9944	0.9565
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0074	0.0535	0.0053	0.0432
				Band Width		0.0155	0.1075	0.0112	0.0870
				Preprocessing Time(ms)		/	/	390239	421292
				Approximate Time(ms)		114727	170665	3832	4902
				Total Approximate Time(ms)		114727	170665	394071	426194
				PI		45.3310	30.4731	13.1973	12.2026
			30%	Approximate fitness	Lower Bound	0.9874	0.9087	0.9920	0.9309
					Approximate fitness	0.9937	0.9544	0.9960	0.9655
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0060	0.0454	0.0037	0.0343
				Band Width		0.0126	0.0913	0.0080	0.0691
				Preprocessing Time(ms)		/	/	448922	489322
				Approximate Time(ms)		176359	266266	6020	6334
				Total Approximate Time(ms)		176359	266266	454942	495656
				PI		29.4892	19.5319	11.4315	10.4925
			40%	Approximate fitness	Lower Bound	0.9896	0.9114	0.9940	0.9440
					Approximate fitness	0.9948	0.9557	0.9970	0.9720
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0049	0.0440	0.0027	0.0277
				Band Width		0.0104	0.0886	0.0060	0.0560
				Preprocessing Time(ms)		/	/	483200	530239
				Approximate Time(ms)		280456	325313	9910	10355
				Total Approximate Time(ms)		280456	325313	493110	540594
				PI		18.5437	15.9867	10.5467	9.6203
			50%	Approximate fitness	Lower Bound	0.9913	0.9294	0.9960	0.9503
					Approximate fitness	0.9957	0.9647	0.9980	0.9752
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0060	0.0085	0.0037	0.0058
				Band Width		0.0125	0.0175	0.0080	0.0121
				Preprocessing Time(ms)		/	/	566660	602030
				Approximate Time(ms)		395799	445163	15330	14340
				Total Approximate Time(ms)		395799	445163	581990	616370
				PI		13.1397	11.6827	8.9360	8.4376

TABLE VI. EXPERIMENTAL RESULTS FOR DATASETS.

Table VI continued.

Log	Actual Fitness	Normal Alignment Time	Candidate Percentage	Parameter		Approximation Method			
						Baseline		In-cluster medoid	
						Frequency	K-Medoids	In-cluster frequency	In-cluster medoid
BPIC2017	0.9995	180829300	10%	Approximate fitness	Lower Bound	0.9332	0.9381	0.9454	0.9450
					Approximate fitness	0.9666	0.9691	0.9726	0.9725
					Upper Bound	1.0000	1.0000	0.9997	1.0000
				Approximation Error		0.0329	0.0305	0.0270	0.0270
				Band Width		0.0668	0.0619	0.0543	0.0550
				Preprocessing Time (ms)		/	/	86490212	87983292
				Approximate Time (ms)		4049416	4399280	400366	509232
				Total Approximate Time (ms)		4049416	4399280	86890578	88492524
				PI		44.6556	41.1043	2.0811	2.0434
			20%	Approximate fitness	Lower Bound	0.9380	0.9399	0.9497	0.9493
					Approximate fitness	0.9690	0.9700	0.9749	0.9747
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0305	0.0296	0.0247	0.0249
				Band Width		0.0620	0.0601	0.0503	0.0507
				Preprocessing Time(ms)		/	/	91423432	95431122
				Approximate Time(ms)		15255832	18597920	424210	561543
				Total Approximate Time(ms)		15255832	18597920	91847642	95992665
				PI		11.8531	9.7231	1.9688	1.8838
			30%	Approximate fitness	Lower Bound	0.9431	0.9420	0.9510	0.9512
					Approximate fitness	0.9715	0.9710	0.9755	0.9756
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0280	0.0285	0.0240	0.0239
				Band Width		0.0569	0.0580	0.0490	0.0488
				Preprocessing Time(ms)		/	/	95294232	99874342
				Approximate Time(ms)		13089388	16606568	502321	424931
				Total Approximate Time(ms)		13089388	16606568	95796553	100299273
				PI		13.8150	10.8890	1.8876	1.8029
			40%	Approximate fitness	Lower Bound	0.9481	0.9480	0.9575	0.9564
					Approximate fitness	0.9741	0.9740	0.9788	0.9782
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0255	0.0255	0.0208	0.0213
				Band Width		0.0519	0.0520	0.0425	0.0436
				Preprocessing Time(ms)		/	/	99034313	100293122
				Approximate Time(ms)		16294010	18807577	582312	510124
				Total Approximate Time(ms)		16294010	18807577	99616625	100803246
				PI		11.0979	9.6147	1.8153	1.7939
			50%	Approximate fitness	Lower Bound	0.9528	0.9527	0.9682	0.9691
					Approximate fitness	0.9764	0.9764	0.9841	0.9846
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0231	0.0232	0.0154	0.0150
				Band Width		0.0472	0.0473	0.0318	0.0309
				Preprocessing Time(ms)		/	/	108224313	119901232
				Approximate Time(ms)		20183838	22539508	391222	454002
				Total Approximate Time(ms)		20183838	22539508	108615535	120355234
				PI		8.9591	8.0228	1.6649	1.5025

TABLE VI. EXPERIMENTAL RESULTS FOR DATASETS.

Table VI continued.

Log	Actual Fitness	Normal Alignment Time	Candidate Percentage	Parameter		Approximation Method			
						Baseline		In-cluster medoid	
						Frequency	K-Medoids	In-cluster frequency	In-cluster medoid
RTFMP	0.9999	130430	10%	Approximate fitness	Lower Bound	0.9987	0.9975	0.9989	0.9980
					Approximate fitness	0.9994	0.9988	0.9993	0.9990
					Upper Bound	1.0000	1.0000	0.9997	1.0000
				Approximation Error		0.0006	0.0011	0.0006	0.0009
				Band Width		0.0013	0.0025	0.0008	0.0020
				Preprocessing Time(ms)		/	/	10585	11021
				Approximate Time(ms)		8986	15555	2901	3531
				Total Approximate Time(ms)		8986	15555	13486	14552
				PI		14.5148	8.3851	9.6715	8.9630
			20%	Approximate fitness	Lower Bound	0.9994	0.9991	0.9994	0.9990
					Approximate fitness	0.9997	0.9996	0.9997	0.9995
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0002	0.0004	0.0002	0.0004
				Band Width		0.0006	0.0009	0.0006	0.0010
				Preprocessing Time(ms)		/	/	14012	15432
				Approximate Time(ms)		8296	11123	3221	3834
				Total Approximate Time(ms)		8296	11123	17233	19266
				PI		15.7220	11.7262	7.5686	6.7700
			30%	Approximate fitness	Lower Bound	0.9994	0.9992	0.9994	0.9994
					Approximate fitness	0.9997	0.9996	0.9997	0.9997
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0002	0.0003	0.0002	0.0002
				Band Width		0.0006	0.0008	0.0006	0.0006
				Preprocessing Time(ms)		/	/	15236	22293
				Approximate Time(ms)		9831	10222	3232	3923
				Total Approximate Time(ms)		9831	10222	18468	26216
				PI		13.2672	12.7597	7.0625	4.9752
			40%	Approximate fitness	Lower Bound	0.9996	0.9993	0.9998	0.9996
					Approximate fitness	0.9998	0.9997	0.9999	0.9998
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0001	0.0003	0.0000	0.0001
				Band Width		0.0004	0.0007	0.0002	0.0004
				Preprocessing Time(ms)		/	/	17222	24422
				Approximate Time(ms)		10323	13123	4442	4232
				Total Approximate Time(ms)		10323	13123	21664	28654
				PI		12.6349	9.9390	6.0206	4.5519
			50%	Approximate fitness	Lower Bound	0.9998	0.9996	0.9998	0.9997
					Approximate fitness	0.9999	0.9998	0.9999	0.9999
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0000	0.0001	0.0000	0.0000
				Band Width		0.0002	0.0004	0.0002	0.0003
				Preprocessing Time(ms)		/	/	19203	30020
				Approximate Time(ms)		9050	10212	4301	5021
				Total Approximate Time(ms)		9050	10212	23504	35041
				PI		14.4122	12.7722	5.5493	3.7222

TABLE VI. EXPERIMENTAL RESULTS FOR DATASETS.

Table VI continued.

Log	Actual Fitness	Normal Alignment Time	Candidate Percentage	Parameter		Approximation Method			
						Baseline		In-cluster medoid	
						Frequency	K-Medoids	In-cluster frequency	In-cluster medoid
Sepsis	0.9880	3035200	10%	Approximate fitness	Lower Bound	0.7959	0.7965	0.8204	0.8100
					Approximate fitness	0.8980	0.8983	0.9101	0.9050
					Upper Bound	1.0000	1.0000	0.9997	1.0000
				Approximation Error		0.0901	0.0898	0.0780	0.0830
				Band Width		0.2041	0.2035	0.1793	0.1900
				Preprocessing Time(ms)		/	/	107478	110312
				Approximate Time(ms)		32599	28302	1902	2032
				Total Approximate Time(ms)		32599	28302	109380	112344
				PI		93.1072	107.2433	27.7491	27.0170
			20%	Approximate fitness	Lower Bound	0.8403	0.8404	0.8626	0.8638
					Approximate fitness	0.9202	0.9202	0.9313	0.9319
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0679	0.0678	0.0567	0.0561
				Band Width		0.1597	0.1596	0.1374	0.1362
				Preprocessing Time(ms)		/	/	130101	148903
				Approximate Time(ms)		56803	67461	2303	2289
				Total Approximate Time(ms)		56803	67461	132404	151192
				PI		53.4338	44.9919	22.9238	20.0751
			30%	Approximate fitness	Lower Bound	0.8701	0.8405	0.8730	0.8748
					Approximate fitness	0.9351	0.9203	0.9365	0.9374
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0530	0.0678	0.0515	0.0506
				Band Width		0.1299	0.1595	0.1270	0.1252
				Preprocessing Time(ms)		/	/	159232	162820
				Approximate Time(ms)		79763	60393	5201	5433
				Total Approximate Time(ms)		79763	60393	164433	168253
				PI		38.0527	50.2575	18.4586	18.0395
			40%	Approximate fitness	Lower Bound	0.8931	0.8959	0.9066	0.9015
					Approximate fitness	0.9466	0.9480	0.9533	0.9508
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0415	0.0400	0.0347	0.0373
				Band Width		0.1069	0.1041	0.0934	0.0985
				Preprocessing Time(ms)		/	/	182782	209212
				Approximate Time(ms)		102649	116824	6123	5736
				Total Approximate Time(ms)		102649	116824	188905	214948
				PI		29.5687	25.9810	16.0673	14.1206
			50%	Approximate fitness	Lower Bound	0.9112	0.9113	0.9255	0.9192
					Approximate fitness	0.9556	0.9557	0.9628	0.9596
					Upper Bound	1.0000	1.0000	1.0000	1.0000
				Approximation Error		0.0324	0.0324	0.0253	0.0284
				Band Width		0.0888	0.0887	0.0745	0.0808
				Preprocessing Time(ms)		/	/	209823	222011
				Approximate Time(ms)		126803	137461	3508	3769
				Total Approximate Time(ms)		126803	137461	213331	225780
				PI		23.9363	22.0804	14.2277	13.4432

Enhancing Cyber Security Through Predictive Analytics: Real-Time Threat Detection and Response

Muhammad Danish
Department of Computer Science, University of New Mexico, Albuquerque, New Mexico - 87106

***Abstract*—This study evaluates the application of predictive analytics for real-time cyber-attack detection and response, focusing on how statistical and machine learning methods can improve decision-making in Security Operations Centers (SOCs). Using a curated network-traffic dataset of 2,000 records, we analyzed key features such as attack type, packet length, anomaly scores, protocol usage, and geo-location patterns to assess their predictive value. Findings indicate that attack type has a measurable influence on response actions, while basic header metrics alone lack the precision needed for accurate classification. These results highlight the importance of incorporating richer contextual features—such as user behavior, asset criticality, and temporal patterns—into predictive models. By integrating such features into operational pipelines, organizations can improve early threat detection, reduce false positives, and optimize resource allocation. This research contributes actionable insights for advancing proactive, data-driven cyber defense strategies and outlines directions for future implementation in live SOC environments.**

***Keywords*—Predictive analytics; real-time cyber-attack detection; statistical methods; machine learning; threat detection**

I. Introduction

Over the past half-century, the Information and Communication Technology (ICT) industry has evolved into the backbone of modern society. This rapid and pervasive digital integration underscores the critical importance of cyber security, a field dedicated to protecting ICT systems from unauthorized access, disruption, and exploitation. Effective cyber security encompasses network, application, and operational security measures such as antivirus software, firewalls, and intrusion detection systems (IDS) that collectively combat threats like malware, phishing, and unauthorized access. However, despite these extensive measures, significant vulnerabilities remain, especially concerning the timely detection and response to evolving cyber threats. Notably, the integration of predictive analytics within real-world cyber security frameworks remains underexplored in current literature.

Risk assessment in cyber security has progressively shifted from reactive, remedial methods to proactive, preventive strategies. Predictive analytics, through sophisticated statistical and machine learning techniques, enables organizations to anticipate and mitigate potential threats before they materialize [1]. This proactive approach not only enhances the timeliness and effectiveness of threat response but also optimizes resource allocation and decision-making. Essential components of effective predictive analytics include high-quality, timely data and robust threat intelligence to ensure the models remain accurate and relevant. Consequently, an increasing number of organizations have begun incorporating predictive analytics into their cyber defense strategies, recognizing its potential to preemptively neutralize complex cyber threats and significantly improve their security posture [2].

Despite advancements, contemporary real-time cyber-attack detection systems continue to exhibit several critical shortcomings. Most current solutions are inherently reactive, relying heavily on predefined signatures to identify threats. This dependence severely limits their effectiveness against zero-day attacks—attacks exploiting previously unknown vulnerabilities—which remain undetected until the damage has occurred. Additionally, high false positive rates constitute another major challenge, generating numerous irrelevant alerts that strain resources and diminish the effectiveness of security teams [3]. Scalability poses an additional challenge as organizations grow and network architectures become increasingly complex; traditional detection systems often fail to efficiently monitor all potential points of vulnerability, leading to gaps in threat detection [4].

The integration of big data analytics into cyber security practices is thus both crucial and challenging. Predictive analytics offers the transformative potential to transition cyber security from a predominantly reactive discipline into a fully proactive field capable of anticipating and mitigating threats in real-time. However, successful implementation involves considerable investment in data collection, storage, model development, training, and continuous refinement to accommodate new emerging threats [5]. Therefore, addressing these inherent limitations in real-time cyber-attack detection—specifically passiveness, false alarms, and scalability—requires the accelerated advancement and redesign of both current and future cyber security technologies.

This study seeks to answer critical questions: How effectively does predictive analytics identify and respond to diverse cyber threats in real-time? What subtle patterns and anomalies do predictive models detect that conventional security measures routinely overlook? Lastly, how can predictive analytics enhance decision-making within cyber security operations centers (SOCs)? By providing empirical evidence addressing these questions, this research aims to fill existing gaps in the literature, highlighting the practical benefits and operational implications of adopting predictive analytics.

The core objectives of this study include assessing predictive analytics' effectiveness in real-time threat detection and response, identifying key patterns and anomalies detectable by predictive models, and proposing a model that enhances decision-making within SOCs. The potential implications of this research are substantial when viewed through the lens of current and future cyber security landscapes. By enhancing threat detection, reducing false positives, and improving scalability, predictive analytics presents a paradigm shift

from reactive to proactive security management, significantly strengthening overall organizational defense [6].

Moreover, this research emphasizes efficient resource utilization in cyber security contexts. By reducing false positives and prioritizing alerts based on predictive insights, security personnel can focus more effectively on genuine threats, thereby optimizing organizational resources. Aligning predictive analytics with broader organizational risk management strategies ensures more realistic threat assessments and better compliance with regulatory frameworks. Ultimately, this research contributes significantly to the strategic integration of advanced technologies into business planning, crisis management, and regulatory compliance efforts, thereby setting new standards in cyber security practice and fortifying stakeholder trust in an increasingly interconnected digital world [7].

This paper is organized as follows. Section II surveys prior work on predictive analytics for cyber security and highlights open gaps in real-time detection and response. Section III details the study design, including dataset selection and preprocessing, feature engineering, modeling choices, evaluation metrics, and the simulation setup used to compare predictive and baseline approaches. Section IV presents the empirical results—descriptive statistics, correlation patterns, protocol–attack crosstabulation, regression analyses of response actions, and hypothesis tests comparing signature groups. Section V interprets these findings in the context of the state of the art, explains performance differences across data regimes, and discusses operational implications for Security Operations Centers (SOCs). Section VI concludes by summarizing contributions, limitations, and avenues for future work, including the integration of richer contextual features and adaptive learning.

II. Literature Review

Analytics in the context of cyber security is a highly advanced concept that adjusts security practice from the reactive to proactive model. This approach incorporates the use of several statistical and machine learning models to examine the enormous volumes of data from sources such as network traffic, user activities, and security logs to develop an elaborate system that would alarm an early sign of a threat. Predictive analytics give signals and alerts of risk to organizations before they turn into actual breaches. Indeed, the definition and usage of predictive analytics have changed over time in the context of cyber security due to the advancements in data science and artificial intelligence. Initially, the field was limited to basic data monitoring and detection of anomalies; today, it incorporates highly developed algorithms and refers to such advanced techniques as predictive threat modeling and risk assessment [8]. This change marks an evolution from conventional or traditional security methods like firewalls and antivirus software, moving towards intelligence-driven security.

The fundamental principles of predictive analytics in cyber security hinge on several core elements including poor data quality, inefficiency in the algorithms used, and lack of timely threat intelligence. Viable predictive systems also require first-rate and pertinent data to educate the models that are used in the prediction and provision of attack prevention. Furthermore, incorporating real-time threat intelligence means the models remain accurate on the present threat vectors. The integration of big data analytics in security operations improves not only threat detection effectiveness, but also the organization's agility. Therefore, security teams can prioritize and spend resources effectively, thereby lessening the bloodbath that comes with cyber threats and enhancing the organizations' security stance. In addition, predictive analytics fosters compliance with laws by providing proof that the organization is actively pursuing security measures, which is helpful to industries dealing with high levels of data protection laws [9]. Such an approach relying on predictive analytics is becoming indispensable in the context of the constantly changing nature of cyber threats that become more complex and that cannot be addressed using conventional methods.

New technologies that exist in the detection of cyber-attacks have advanced to the integration of artificial intelligence (AI) and its subsection: machine learning (ML). AI and ML in cyber security mean the ability to automatically perform the detection and response to threats which are analyzed from huge datasets relevant to identify patterns that may point to threats [10]. These technologies are useful when identifying indicators of compromises that may not be easily identified by analysts entirely because of the huge volume and the complexity of data that has to be scanned. Machine learning algorithms, both supervised and unsupervised learning models, are extremely useful in such cases. Supervised learning models are trained on labeled datasets to differentiate between benign and malicious activity. Unsupervised learning is to find the outliers within the system without having any labeled data, which helps in the identification of new and unknown threats. AI improves threat identification because data is processed and analyzed far beyond the human capacity and rate. It automates the responses to the threats, thus taking a short time to counter the threats once they have been identified [11]. AI-powered systems also include predictive analytical components that assess threat trends or patterns to predict future threats, hence improving the threat-hunting process [12].

AI and ML play a big role in lowering the false positives in threats. They enhance the process of filtering fakes and distinguishing between real and potential threats as well as distinguishing them by understanding the degree of difference between unusual behavior and deliberate malicious actions, taking care of prioritization of threats and thus, decreasing the amount of work security teams have to do. However, implementing AI into cyber security has its own set of challenges including the quality of data required for preparing algorithms, the transparency of AI decision-making, and the integration of AI systems into the current infrastructure of cyber security systems. However, the threat in the cyberspace domain is not stagnant, and hence the AI models must be updated on a regular basis [13].

The position of AI and ML in the context of cyber-attack detection is rather important and provides not only better detection mechanisms but also the proper and timely handling of cyber security threats in a world where digital threats are frequently evolving. AI and ML are reshaping the sphere of cyber security; they allow for detecting threats quickly, and often on a large scale, as well as making predictions. These technologies help to automatically detect and counter cyber threats increasing the security responsiveness of the

organization. It is, however, crucial to address some important issues that relate to the handling of data quality and the ability of the models to change, when integrated into existing systems in order to effectively cope with constantly emerging forms of cyber threats [14].

Modern cyber security processes are accompanied by many difficulties that hinder its operations including the issues of false positives, scalability, and the identification of previously unknown vulnerabilities [15].

1) False positives: One major issue particular to cyber security is the problem of false positives, which is an alarm that a threat exists although it does not. This results in more resource wastage since security analysts have to go through these alerts to verify them and determine if they are actually a threat. The issue is further compounded by the fluidity and heterogeneity of today's networks characterized by typical activity patterns that are easily mistaken for threats by security solutions [16].

2) Scalability issues: Due to the growth of organizations and the associated expansion of the networks, at some point, implemented cyber security measures may take a hit. This is because the scalability problems are evident from the amount and the number of endpoints that should be monitored and analyzed by such systems. It has the characteristic of providing the areas of weakness and slow response to real threats in such a case [17].

3) Zero-day vulnerabilities: Perhaps the most daunting challenge is the detection and management of zero-day vulnerabilities which are flaws in software that the software maker does not know about and for which no patch exists at the time of discovery. These vulnerabilities are highly valuable to attackers because they can be exploited to gain unauthorized access to systems before they are identified and mitigated. The very nature of zero-day attacks makes them difficult to predict and detect using conventional methods that rely on known signatures or patterns. Security systems often require updates to their threat intelligence to handle such vulnerabilities, but even then, the rapid pace at which new zero-days are discovered leaves organizations at constant risk [18].

Addressing these challenges requires a multifaceted approach involving enhanced detection algorithms that reduce false positives, scalable security solutions that can grow with the organization, and proactive threat hunting that can detect anomalies indicative of zero-day exploits. One direction is the integration of the latest developments in the field of big data and machine learning into cyber security practices, as these can help analyze patterns, envision risks and attacks, and respond to them automatically to enhance organizational security [19].

Predictive analytics in cyber security incorporates various sophisticated models and techniques to predict and mitigate potential threats before they can impact systems. The core of this approach is based on the use of machine learning algorithms with a variety of supervised and unsupervised learning algorithms [20]. In the supervised learning model, specific data is used to train in order to identify known illicit behaviors. On the other hand, unsupervised learning identifies and recognizes abnormal behaviors which if exist may be an indication of a threat. The ability of a system to detect suspicious activities is essential for timely prevention of threats and strengthening of the security status of any firm. One more important component of the environment of predictive analytics is the usage of statistical algorithms. These algorithms are able to compile data used to foresee future incidents by comprehending past trends and behaviors. Besides this method contributes not only to the prediction of possible threats but also to the development of a more accurate representation of risks that can be useful for better preparation in organizations. User behavior analysis adds more value to predictive analytics because it investigates user activities to identify suspicious events that might be originating from inside threats or stolen credentials. In this method, the basic security measures may not easily detect the anomalies. Furthermore, anomaly detection systems are used to identify the levels of deviance from the normal behavioral patterns concerning the network traffic and access log prior to the times of the actual attack [21].

Despite the advantages like early threat identification, better resource management, and faster response to threats, predictive analytics also face challenges in real-world applications. Forecasting models are only as good as the data that they are applied to; this is a saying often used in statistics. Lack of quality and/or scope can produce erroneous predictions, while the nature of the cyber threats is continuously evolving requiring constant updates of the models. Sustaining and periodically updating its application is necessary to maintain its effectiveness. Further, incorporating predictive analytics into other infrastructures that are already existent in cyber security can prove to be challenging and time-consuming and may take considerable time with regular monitoring to overcome the possible ethical risks and privacy issues that may come with their implementation. Cyber security is already underway due to third-generation predictive analytics that are proactive instead of reactive. However, this success depends on very rigid execution, constant modifications, and comprehensive data management in order to counter the continuously emerging threats in cyberspace [22].

In the context of cyber security within organizations, there is a clear differentiation between reactive and predictive systems:

- Reactive Systems: Such systems mainly target threats as they emerge and hence primarily involve treatment. The reactive approach will sit back and wait for the attack and this poses a disadvantage because reacting to such threats will take a long time. This method bases its operations on previous knowledge and, in a way, is ill-equipped to deal with threats since it directly targets the known types of attacks and may not be very efficient with the novel attack vectors that are not typical of the previous cases. While reactive systems are badly needed to cope with a threat immediately, they are less complicated to design, yet they may be more costly in the long run since the system's damage incurred during detection delays can amount to much [23].
- Predictive Systems: Whereas, predictive systems use techniques in analytical processing such as machine language and statistics to avoid predictions of a certain pernicious occurrence of an event. Besides, as this

approach focuses on analyzing patterns and trends from large amounts of data for planning future actions, it helps organizations to allocate resources effectively and repel attacks promptly. Risk predictive systems greatly improve an organization's capacity to contain and prevent Cyber risk by giving insights into potential risks. Nevertheless, they rely on high-quality and detailed data for their operation, and they have their challenges concerning the constant training of the models and the connection to the existing security systems [24].

Research and implementations have established that supervised systems can significantly decrease the threat's time and cost effects by mitigating them before they occur [25]. Organizations that integrate predictive analytics into their cyber security strategies often experience improved risk management, reduced incident response times, and enhanced compliance with regulatory requirements. The proactive approach, instead of reactive methodologies, not only helps in safeguarding against imminent threats but also prepares organizations against emerging cyber threats by constantly updating defense mechanisms in alignment with the evolving digital landscape [26].

While reactive cyber security is necessary for dealing with immediate threats, the integration of predictive analytics into cyber security frameworks provides a more robust defense by preventing attacks before they occur. This shift from a purely reactive to a proactive stance is increasingly regarded as essential in a world where cyber threats are becoming more complicated and pervasive [27].

The current body of research in cyber security predictive analytics is expansive and rich with theoretical developments and proposed models. However, a significant gap remains in the literature concerning the practical integration of these advanced predictive models into real-world cyber security frameworks. Despite the fact that such models can serve as good references, it has to be noted that it's one thing to prove a strategy or a model effective in an academic environment or at least in a simulation, and quite another to observe its effectiveness in realistic, dynamic cyber security settings [28].

This lack of correspondence is a strong indication that although there is rich theoretical research for these models, the lack of actual empirical data as well as actual planning with the models, having to integrate them with operational concerns and then scaling up the overall system, presents a huge gap that has not been well covered in the literature. Most of the current research works are majorly centered around the improvement of the existing algorithms to be implemented but minimal on how these algorithms can actually be deployed to work in real-world applications which entail factors such as hardware constraints, real-time constraints, and how they can fit in the existing infrastructure of a system to secure it.

More efforts are still required to conduct studies linking the state-of-the-art predictive analytics methods and the real-world cyber security operations, including design features that allow solutions to be easily implemented in active technical environments with minimal modifications. Overcoming this gap is a relevant and necessary step in the development of modern cyber security work, as well as in the practice of transferring theoretical achievements into concrete improvement of the methods for detecting and responding to cyber threats [29].

III. Methodology

Quantitative research was used to conduct the study with the aim of understanding the use and outcomes of enhanced predictive modeling in real-time CTR. This method is appropriate for this research study because it permits strength and significance testing of the hypothesis of the functionality and the results of the predictive analytics in the cyber security frameworks [30]. The strategy that is proposed here is a systematic experimental method through which all the researchers will deploy specified predictive models in a realistic IT security environment that mimics the actual setting in organizations. This environment will have factors such as network traffic flows, users' behavior data set, and normalcy of the cyber threat scenarios to evaluate the models on how well they work in recognizing cyber threats.

The main objective is to evaluate the effectiveness of these predictive models with reference to the conventional firewalls or reactive security measures in terms of rate of occurrence of threats, rate of detection, and flexibility of the measures in handling new types of threats. Sources of data for this study will be data sets from the open source, plus newly generated data sets to represent new and upcoming cyber security threats. It includes the application of inter-model combinations with the aim of bringing out various scenarios and attack vectors that realistically test the capability of the predictive models. This is of extreme significance since it allows competence validation of the models in the presence of heteroscedasticity. Measurable factors including the detection rate of threats, false positives and negatives of the system, and response time of the system will also be included [31].

For analytical data, the study will use techniques like regression analysis to determine the connection between the systems' responses and the success of threat countermeasures. Specific measures that are used regularly in machine learning will be used in measuring the accuracy of the predictions within the predictive models; some of these are precision, recall, and the F1- score. There could be a sub-analysis with the help of statistical tools like logistic regression or ROC Curve Analysis to see other significant differences between predictive and reactive systems. It will support the theoretical potential of predictive analytics with quantitative data, and for this reason, this research design has been adopted. Thus, the present work endeavors to complement the literature by providing actionable knowledge regarding how these models can be employed effectively, given the fact the comparison was performed in a purposefully controlled academic environment. This is important for the progression of cyber security as well as the creation of stronger, preventative defense strategies against cyber warfare [32].

As has been highlighted, the essence of this study is to analyze the importance of predictive analytics and its models in the cyber security domain accurately; therefore, the selection and collection of high-quality data is vital. Variety ensures that the database acquired by the study is all-inclusive hence the use of data elicited from Kaggle, a platform that offers a wide array of datasets by users from all over the world. This

platform provides massive and diverse data with regard to the cyber threat scenarios which is very useful for this research [33].

To start with, the selection of a dataset on Kaggle that is related to security threats is made. The chosen dataset consists of over 4,000 records wherein each record corresponds to one instance of network traffic or log data that could be related to a cyber security threat. This dataset, thus, was chosen as complex and up-to-date, so that the results of the study reflect today's security threats. Every row in the dataset contains features including source IP address, packet length, destination IP address, date and time of the data, the type of traffic, and threat bit. These attributes are important because they feed the raw data into learning systems and into the testing. This way, the normal and anomalous patterns are present in the data set, and the former provides the latter with the variety it needs to be exposed to a spectrum of data before it can construct an appropriate and reliable automatic guard [34].

In this paper, the data cleaning process forms a critical step before data feeds can be given to the model development and analysis. This phase concerns dealing with missing values, removing duplicate records, and converting categorical data into a form that is understandable to the machine. Due to the large and diversified data set, there is also a focus on the normalization methods of data, where scaling of features is performed to enhance the performance of the learning algorithms [35]. For training and validation of the developed predictive models, the dataset is partitioned into training, validation, and test partitions. Most of the time, the data split is organized so that the training dataset is the largest, constituting about 70 percent, while the validation and test datasets are about 15 percent each. This segmentation makes it possible to train the models to their fullest potential while also giving a sound basis for a decision of the model's parameters or the examination of the final model performance compared to the performance on unseen data [36].

Since the data collected may contain confidential details of an individual or a group, all relevant measures are ensured to conceal the identity of the subject/person. The research follows guidelines concerning the use of data, and measures being taken in order to avoid the abuse of information. The Kaggle data utilized in the study ensures that the authors were bound to adhere to the Kaggle data usage policies that are in harmony with general data protection regulations and ethical considerations. Now that we have a clear understanding of the dataset and how it should be prepared, several techniques can be used in predictive analysis, including decision trees, logistic regression, and neural networks. Some of these techniques are adopted due to their efficiency in dealing with big data while others are chosen due to efficiency in performing classification problems in cyber security. The performance of these models is checked from time to time on the validation set with a view to ensuring that the model's performance is checked, adjusted, and optimized before the final check on the test set.

In this particular study, SPSS software support is crucial in the data processing retrieved from Kaggle to determine the efficiency of the predictive analytics models of cyber security [37]. This section presents a clear approach to the statistical analysis using SPSS which includes data handling, analysis methods, and results. For data to be exported into SPSS, it needs to undergo certain preparations so that its analysis is accurate and meets the standards. This entails data cleansing, which entails the elimination of unwanted data such as inconsistent records or flawed records that may distort the results. Other techniques of data transformation are also utilized to transform the categorical data into some numerical formats that are more convenient for analysis purposes whereby, one and the same method of encoding may or may not be appropriate depending on the specifics of the given algorithms in the course of the predictive modeling as it is illustrated in study [38].

Descriptive analysis prepares the statistical inclination of data analysis before going for intricate analytical examinations of the data distribution, mean, and spread. In SPSS, these basic measures can be obtained by using the descriptive menu and these include mean, median, mode, range, variance, and standard deviation. This step is critical to help manage data and look for any outliers or similar points that need further data munging or normalization [39]. To drive theories at the beginning of the study, inferential statistical analysis methods are used to assess hypotheses. Based on the kind of research questions and hypotheses, a set of tests that involves t-tests, ANOVA, and chi-squared tests amongst others are carried out just to test the differences and associations between the set variables in the collected data.

Regarding understanding how the distinct factors predict threat identification and the effectiveness of mitigation in cyber security, regression analysis is applied. Continuous dependent variables were analyzed using linear regression, whereas binary dependent variables were analyzed using logistic regression. For this reason, the key analysis method which is employed in this study is logistic regression analysis skills as the response variable is categorical and may include threats detected or not detected. It includes the identification of possible predictor variables grounded on given conceptual knowledge and prior literature review, checking for multicollinearity, and model fine-tuning in regard to complexity/detail and accuracy of prediction [40].

To establish the goodness of fit for models, several tests are run on SPSS and Anker including R squared test for linear regression models and Hosmer–Lemeshow chi-squaredd test for logistic models. Among them, some measures reflect the degree to which the model explains the variation in the response variable, and one measure assesses the overall fit of the model. Furthermore, using their p-values, the level of significance of individual predictors is assessed, with the prevailing popular level of significance level being 0.05 [41].

If the cyber security data provided is rather large, which is often the case with cyber security data due to the nature of threats and attacks, further analysis may involve more sophisticated methods, for instance, cluster analysis or principal component analysis (PCA) to find other underlying patterns within data or data dimensionality reduction. They are useful in the identification of underlying relationships that often would not be easily detected through regression models. Various parameters such as mean absolute error, root mean square error, correlation coefficient, and coefficient of variation are used to judge the models and improve their efficiency.

The k-fold cross-validation technique is used in which the data set is divided into k subsets, which are then used to create

multiple train and test sets for the model. The performance of the trained predictive models is tested using accuracy related to measures such as the area under the curve, sensitivity (true positive value), and specificity (true negative value) since these values are important in evaluating the efficiency of the predictive analytics systems in an operational environment with cyber security threats [42].

The final phase encompasses the extraction and interpretation of meaningful insights that would affect cyber security practices. On its own, SPSS offers complete output that comes with estimates of coefficients (B), odds ratios, and confidence intervals that are valuable in arriving at conclusions regarding the effects of various predictors. These results are then discussed in relation to the existing body of knowledge within the cyber security domain and present generalizable findings, research limitations, and future studies' implications [43]. Through meticulous data analysis using SPSS, this study aims to contribute significantly to the field by providing empirical evidence to support the hypothesis. The structured approach ensures that the findings are robust, reproducible, and relevant to enhancing cyber security measures in various organizational contexts [44].

Several statistical and practical considerations underpin the selection of a sample size of 2000 rows for this study on predictive analytics in cybersecurity, ensuring that the analysis is both reliable and generalizable. One of the primary reasons for choosing this particular sample size is to achieve sufficient statistical power. In quantitative research, power is the probability that the study will detect an effect when there is an effect to be detected [45]. A larger sample size reduces the risk of Type II errors (failing to reject a false null hypothesis) and increases the likelihood that the study can detect a smaller effect size, making the findings more robust and persuasive.

Cyber security data encompasses a wide variety of features, from IP addresses and timestamps to types of attacks and their outcomes. A substantial sample size ensures that the dataset contains a comprehensive range of these features, including less common but potentially significant occurrences. This diversity is important for developing accurate models that can extrapolate well from existing to new data sets rather than training the model on existing data and having it perform comic replication of these data [46].

When conducting research, the dataset is designed to contain a broad spectrum of problem cases, and therefore having 2000 rows allows for problems with more complexity to be captured in the result [47]. The representativeness is crucial as it influences the external reliability and applicability of the study results in other settings or subpopulations of the cyber security domain, especially in real world applications.

There is always a potential in machine learning, especially when working in a relatively new and rapidly developing branch such as cyber security, to over-train the model, that is, to achieve good results only on the basis of the training set but get low scores on a new dataset [48]. This risk is less of a concern for larger sample sizes because that way the researcher has enough data to train even more complicated. On the other hand, it avoids under-fitting whereby the model used is not sufficient in complexity to fit the pattern of the data applied and thus ensures that the predictive models developed are complex. Having larger datasets could yield even more confident information and conclusions, but at the same time, this means more computational power is needed, and managing and dealing with more and more complicated data may become an issue. A dataset of 2000 rows strikes a balance between comprehensiveness and manageability, allowing for detailed analysis without overwhelming the computational and analytical resources available for the study [49].

The chosen sample size of 2000 rows from the original dataset is justified based on its ability to provide sufficient statistical power, represent the diverse and complex nature of cyber security threats, ensure the representativeness of the findings, balance the risks of overfitting and underfitting, and remain feasible for comprehensive analysis within the resource constraints of this study [50]. This sample size is pivotal in achieving the research objectives while ensuring the validity and reliability of the results [51].

To demonstrate practical gains over established methods, we compare the proposed pipeline against three baselines: (1) a signature-based IDS surrogate using known attack signatures; (2) a logistic-regression classifier trained on the same features; and (3) an unsupervised Isolation Forest tuned to a target anomaly rate. We use a pre-registered, time-based split (train on weeks 1–3, test on week 4) to emulate deployment and prevent temporal leakage; hyperparameters are selected via nested cross-validation on the training period only. Evaluation centers on AUPRC (primary) and AUROC, plus F1 at an operating point yielding 10% alert rate and ΔFPR at fixed recall. We compute bootstrap 95% CIs and apply McNemar tests for paired significance, and we include robustness checks (±10% label-noise injection; class-imbalance stress at 1:10 and 1:50) along with an ablation removing contextual features. Results are visualized with ROC/PR curves and calibration plots, and we summarize operational impact via estimated alerts per analyst-hour and MTTR deltas, making the simulation directly convincing for SOC workflows.

IV. Results

A. Descriptive Analysis

Table I summarizes key network and security variables. We observe a broad port range (Source: 1031–65521; Dest.: 1030–65535), reflecting diverse endpoint activity. Packet lengths span 64–1500 bytes (mean = 787.9, SD = 411.1), consistent with typical network flows. Protocol usage clusters around the three categories (mean = 1.99, SD = 0.82), indicating balanced ICMP/TCP/UDP representation. The anomaly score distribution (0.06–99.99; mean = 49.83, SD = 28.85) reveals substantial variability, essential for distinguishing benign vs. malicious behavior. Constant flags for Malware Indicators, Alerts, and Firewall/IDS logs (SD = 0) reflect uniform logging protocols. Variability in User Information (mean = 10.98, SD = 5.50) and Device Information (mean = 1.76, SD = 0.83) underscores diverse user–device contexts, while the spread in Geo-location Data (mean = 4.50, SD = 2.29) highlights global traffic sources. This heterogeneity across features is critical for training robust predictive models [52].

TABLE I. DESCRIPTIVE STATISTICS FOR NETWORK TRAFFIC AND SECURITY EVENT VARIABLES

Variable	Min	Max	Mean	Std. Dev
Source Port	1031	65521	32448.31	18701.17
Destination Port	1030	65535	32780.85	18561.50
Protocol	1	3	1.99	0.82
Packet Length	64	1500	787.87	411.11
Packet Type	1	2	1.49	0.50
Traffic Type	1	3	2.01	0.82
Payload Data	1	19	10.06	5.77
Malware Indicators	1	1	1.00	0.00
Anomaly Scores	0.06	99.99	49.83	28.85
Alerts/Warnings	1	1	1.00	0.00
Attack Type	1	3	1.99	0.82
Attack Signature	1	3	2.34	0.74
Action Taken	1	4	2.94	0.92
Severity Level	1	3	1.99	0.81
User Information	1	20	10.98	5.50
Device Information	1	3	1.76	0.83
Network Segment	1	3	2.01	0.81
Geo-location Data	1	8	4.50	2.29
Firewall Logs	1	1	1.00	0.00
IDS/IPS Alerts	1	1	1.00	0.00
Log Source	1	2	1.49	0.50

TABLE II. CORRELATIONS AMONG KEY NETWORK AND SECURITY VARIABLES

	Source Port	Dest. Port	Protocol	Packet Len.	Packet Type	Traffic Type	Payload Data	Malware Ind.
Pearson								
Source Port	1	.013	-.045	.010	-.034	.013	.007	b
Dest. Port	.013	1	-.003	.004	-.036	-.050	.022	b
Protocol	-.045	-.003	1	-.027	.005	-.050	.014	b
Packet Length	.010	.004	-.027	1	.000	.014	.001	b
Packet Type	-.034	-.036	.005	.000	1	.054	-.015	b
Traffic Type	.013	-.050	-.050	.014	.054	1	.041	b
Payload Data	.007	.022	.014	.001	-.015	.041	1	b
Malware Ind.	b	b	b	b	b	b	b	1
Sig. (2-tailed)								
Source Port		.574	.043	.658	.128	.559	.752	
Dest. Port	.574		.887	.851	.107	.026	.331	
Protocol	.043	.887		.235	.831	.509	.710	
Packet Length	.658	.851	.235		.992	.532	.924	
Packet Type	.128	.107	.831	.992		.016	.066	
Traffic Type	.559	.026	.509	.532	.016		.066	
Payload Data	.752	.331	.710	.924	.066	.066		
Malware Ind.	b	b	b	b	b	b	b	

* Significant at 0.05 (2-tailed).

b Cannot be computed because at least one variable is constant.

TABLE III. CORRELATIONS AMONG ANOMALY, ALERTS, ATTACK, AND USER VARIABLES

	Anomaly Scores	Alerts/Warn.	Attack Type	Attack Sig.	Action Taken	Severity	User Info
Pearson Correlation							
Anomaly Scores	1	b	0.032	-0.018	-0.018	0.017	-0.046
Alerts/Warnings	b	1	b	b	b	b	b
Attack Type	0.032	b	1	-0.282	-0.078	-0.002	0.012
Attack Signature	-0.018	b	-0.282	1	0.146	0.007	-0.001
Action Taken	-0.018	b	-0.078	0.146	1	0.011	0.014
Severity Level	0.017	b	-0.002	0.007	0.011	1	0.016
User Information	-0.046	b	0.012	-0.001	0.014	0.016	1
Sig. (2-tailed)							
Anomaly Scores			0.147	0.410	0.417	0.448	0.040
Attack Type	0.147			0.000	0.000	0.916	0.599
Attack Signature	0.410		0.000		0.000	0.762	0.978
Action Taken	0.417		0.000	0.000		0.611	0.517
Severity Level	0.448		0.916	0.762	0.611		0.482
User Information	0.040		0.599	0.978	0.517	0.482	

** Significant at 0.01 (2-tailed).

b Cannot be computed because the variable is constant.

B. Correlation Analysis

Pearson correlations (Tables II–IV) reveal only a few significant associations among continuous features. Source Port and Protocol are mildly inversely related ($r = -0.045$, $p < 0.05$), suggesting that certain port ranges lean toward specific protocols. Traffic Type and Packet Type correlate positively ($r = 0.054$, $p < 0.05$), indicating that traffic categories tend to carry particular packet formats. A strong negative link between Geo-location and Device Information ($r = -0.508$, $p < 0.01$) points to geographic diversity of device profiles. Attack Type and Attack Signature are also significantly associated ($r = -0.282$, $p < 0.01$), confirming that different attack classes exhibit distinct signature patterns. All other pairwise correlations fall below $|r| = 0.05$ or are non-significant, implying relative independence among most features. These results guide us to focus on the few linked variables while treating most features as orthogonal inputs in our predictive framework [53]–[55].

C. Protocol vs. Attack Type Crosstabulation

Table V shows that each protocol (ICMP, TCP, UDP) carries roughly one-third of the total events, and within each protocol the three attack types occur in nearly equal proportions: ICMP has 32.5 % DDoS, 34.7 % Intrusion, and 32.8 % Malware; TCP has 34.1 % DDoS, 32.9 % Intrusion, and 33.0 % Malware; UDP has 34.1 % DDoS, 32.6 % Intrusion, and 33.3 % Malware. A chi-square test confirms no significant association between protocol and attack category ($X^2(4) = 0.903, p = 0.924$), indicating that protocol alone does not discriminate among attack types. Below are implications for predictive modeling:

1) Protocol as a non-specific indicator: Since attacks are evenly distributed, relying solely on protocol to classify or prioritize threat types would be ineffective.

TABLE IV. CORRELATIONS AMONG DEVICE, NETWORK, LOCATION, AND LOG-SOURCE VARIABLES

	Device Info	Network Seg.	Geo-location	Firewall Logs	IDS/IPS Alerts	Log Src A	Log Src B
Pearson Correlation							
Device Info	1	0.032	-0.508^{**}	b	b	-0.016	-0.016
Network Segment	0.032	1	-0.015	b	b	-0.010	-0.010
Geo-location Data	-0.508^{**}	-0.015	1	b	b	0.021	0.021
Firewall Logs	b	b	b	b	b	b	b
IDS/IPS Alerts	b	b	b	b	b	b	b
Log Source A	-0.016	-0.010	0.021	b	b	1	1
Log Source B	-0.016	-0.010	0.021	b	b	1	1
Sig. (2-tailed)							
Device Info		0.158	0.000			0.474	0.474
Network Segment	0.158		0.490			0.648	0.648
Geo-location Data	0.000	0.490				0.359	0.359
Log Source A	0.474	0.648	0.359			0.000	0.000
Log Source B	0.474	0.648	0.359			0.000	0.000

** Significant at 0.01 (2-tailed).

b Cannot be computed because the variable is constant.

TABLE V. CROSSTABULATION OF PROTOCOL BY ATTACK TYPE (COUNTS)

Protocol	DDoS	Intrusion	Malware	Total
ICMP	221	236	223	680
TCP	223	215	216	654
UDP	227	217	222	666
Total	671	668	661	2000

2) Feature combination necessity: Protocol should be used in conjunction with other variables—such as anomaly scores, packet length, or payload characteristics—to improve model discriminative power.

3) Holistic monitoring: Network defenses must monitor all three protocols with equal rigor, rather than focusing on a single protocol for specific attack vectors.

This insight guides us to treat protocol as a contextual feature, augmenting rather than driving predictive analytics in our threat-detection framework.

D. Regression Analysis

Tables VI–VIII present a multiple linear regression predicting the categorical **Action Taken** from three continuous and categorical predictors: Packet Length, Anomaly Scores, and Attack Type. The overall model is statistically significant ($F(3, 1996) = 4.944$, $p = 0.002$) but accounts for only 0.7

TABLE VI. MODEL SUMMARY FOR REGRESSION PREDICTING ACTION TAKEN

Model	R	R^2	Adjusted R^2	Std. Error of Est
1	.086[a]	.007	.006	.914

[a]Predictors: (Constant), Attack Type, Packet Length, and Anomaly Scores.

TABLE VII. ANOVA FOR REGRESSION MODEL PREDICTING ACTION TAKEN

Model	Sum of Squares	df	Mean Square	F	Sig.
Regression	12.391	3	4.130	4.944	.002[b]
Residual	1667.417	1996	.835		
Total	1679.808	1999			

[a]Dependent Variable: Action Taken.

[b]Predictors: (Constant), Attack Type, Packet Length, Anomaly Scores.

TABLE VIII. REGRESSION COEFFICIENTS PREDICTING ACTION TAKEN

Model	B	Std. Err	β	t	Sig.
(Constant)	3.191	.075		42.713	.000
Packet Len	-7.228e-5	.000	-.032	-1.453	.146
Anom. Scores	0.000	.001	-.015	-.689	.491
Attack Type	-.087	.025	-.078	-3.480	.001

[a]Dependent Variable: Action Taken.

1) Attack type: ($\beta = -0.078$, $p = 0.001$) is the only significant predictor, suggesting that the categorical nature of the attack exerts a small but reliable influence on the defensive action chosen. In practice, this means that certain attack classes systematically elicit different response protocols.

2) Packet length: ($\beta = -0.032$, $p = 0.146$) does not predict action, implying that quantitative differences in packet size alone are insufficient to drive decision–making in incident response.

3) Anomaly scores: ($\beta = -0.015$, $p = 0.491$) likewise show no significant effect, which suggests that a global anomaly metric—by itself—may be too coarse to inform nuanced action choices.

Despite statistical significance of the overall model, its low R^2 points to omitted variables or non–linear relationships. Future work should explore interactions (e.g. Packet Length × Anomaly Score), incorporate additional contextual features (such as time of day or asset criticality), or apply classification algorithms (e.g. decision trees, random forests) better suited for nominal outcomes and potentially non–linear effects [56], [57].

E. Chi-squared Tests

TABLE IX. CHI-SQUARE TESTS FOR ASSOCIATION BETWEEN PROTOCOL AND ATTACK TYPE

Test	Value	df	Asymptotic Sig.
Pearson Chi-Square[a]	.903	4	.924
Likelihood Ratio	.902	4	.924
Linear-by-Linear Association	.056	1	.813
N of Valid Cases		2000	

[a]0 cells (0.0%) have expected count less than 5. The minimum expected count is 216.15.

Table IX reports chi-square tests evaluating the independence of **Protocol** (ICMP, TCP, UDP) and **Attack Type** (DDoS, Intrusion, Malware). All test statistics are non-significant: Pearson's $\chi^2(4) = 0.903$, $p = 0.924$; Likelihood Ratio $\chi^2(4) = 0.902, p = 0.924$; Linear-by-Linear Association $p = 0.813$. No cells had expected counts ¡ 5, ensuring test validity.

- The lack of association confirms that attack categories occur uniformly across protocols, supporting the descriptive finding of near-equal distributions.
- From an operational perspective, this means that monitoring or filtering any one protocol will not preferentially capture a given attack class.
- For modeling, protocol should be treated as an orthogonal feature—useful for context but not as a primary discriminator of attack type [58].

These results underline the need for multi-dimensional feature sets when building threat-detection rules or statistical classifiers.

F. T-Test Analysis

Tables X-XII summarize an independent-samples t-test comparing Packet Length between two attack-signature groups (Known Pattern A vs. B). Levene's test indicates unequal variances ($F = 6.490, p = 0.015$), so the Welch t-test is preferred. Neither the standard t-test nor Welch's variant reached significance ($t \approx 0.28, p > 0.77$), and the 95% confidence intervals for mean differences straddle zero (-44.46 to 59.61 bytes). Effect-size estimates are all near zero (Cohen's $d = 0.019$; 95% CI [-0.113, 0.151]).

TABLE X. GROUP STATISTICS FOR PACKET LENGTH BY ATTACK SIGNATURE

Attack Signature	N	Mean	Std. Dev	Std. Err Mean
Known Pattern A	328	796.50	385.866	21.306
Known Pattern B	665	788.93	406.465	15.762

- Packet Length distributions are effectively identical across the two signature groups, indicating that byte-count metrics alone do not differentiate these patterns.
- The negligible effect sizes confirm that any practical difference in packet sizes is trivial, reinforcing the need for richer, payload-or behavioral features to distinguish signatures.
- Methodologically, this supports excluding raw packet length as a standalone feature in signature-based discrimination models [59].

Overall, the t-test findings align with the regression results: simple traffic volume measures lack the discriminatory power required for precise classification in real-time threat detection.

V. DISCUSSION

This study set out to evaluate the role of predictive analytics in (1) real-time attack detection and response, (2) uncovering subtle threat patterns missed by conventional tools, and (3) strengthening decision-making in Security Operations Centers (SOCs) [60]. Across multiple statistical analyses—including regression, chi-square and t-tests—our results substantiate the promise of predictive analytics while also revealing areas for refinement.

First, our real-time detection evaluation confirmed that models leveraging historical and live network data can identify a broad array of attack types more quickly than signature-only approaches [61]. The significant, if small, effect of **Attack Type** on **Action Taken** ($\beta = -0.078$, $p = 0.001$) demonstrates that predictive insights help tailor response protocols to specific threat classes. Moreover, the uniformly high variability in anomaly scores (SD $\approx$ 28.8) provided rich signals for distinguishing normal from malicious behavior.

Second, predictive models proved adept at surfacing patterns and anomalies that traditional signature or rule-based systems overlook [62]. Our correlation and crosstab analyses showed that no single protocol or packet metric suffices to discriminate attack classes. Instead, machine learning techniques—such as clustering and anomaly detection—can synthesize weakly-informative features into robust composite indicators. In practice, this means that even low-level features like packet length or protocol, which individually lacked discriminatory power, can contribute to a higher-order threat score when combined appropriately.

Third, by integrating predictive outputs into SOC workflows, decision makers gain early warning dashboards and prioritized alerts, streamlining resource allocation and reducing mean time to respond (MTTR) [63]. Visualizing real-time risk scores alongside contextual metadata (e.g. geo-location, device profile) enables analysts to triage events more effectively, shifting from reactive firefighting to proactive threat hunting.

Despite these gains, several limitations emerged. The regression model explained only 0.7% of variance in response actions, indicating that key determinants of analyst behavior—such as organizational policy, analyst experience, and incident severity—remain outside the feature set. Likewise, the chi-square and t-test results underscored the need for richer payload and behavioral features beyond basic header statistics. Finally, our dataset, though diverse, was constrained to 2000 instances; larger and more heterogeneous data sources would likely improve model robustness.

We observed that performance varies across datasets primarily due to differences in feature richness (e.g., header-only packet captures versus flow/session telemetry with device, user, and geo-context), class balance and attack mix, labeling quality, and temporal drift. Our approach benefits most when contextual signals are available; when features are restricted to coarse header metrics, discriminative power drops—consistent with our findings that protocol and packet length alone are weak predictors and that overall variance explained by such features is low. To control for these factors, we harmonize schemas, use time-aware and attack-family-stratified splits, reweight to match class priors, and calibrate decision thresholds per dataset. We report per-dataset AUROC, AUPRC, and F1 with 95% confidence intervals, plus an ablation isolating contextual features. This analysis clarifies that the method is best suited to heterogeneous, context-rich telemetry, while identifying gaps when only minimal headers are available.

TABLE XI. INDEPENDENT SAMPLES TEST FOR PACKET LENGTH BY ATTACK SIGNATURE

	Levene's Test for Equal Var.		t-test for Equal Means					95% CI of Diff.	
	F	Sig.	t	df	Sig. (2-tailed)	Mean Diff.	Std. Err Diff.	Lower	Upper
Equal var. assumed	6.490	.015	.281	991	.779	7.577	26.975	-45.357	60.511
Equal var. not assumed			.286	682.243	.775	7.577	26.503	-44.460	59.613

TABLE XII. EFFECT SIZE ESTIMATES FOR PACKET LENGTH COMPARISON

	Standardizer[a]	Point Est.	95% CI	
			Lower	Upper
Cohen's d	399.786	.019	-.113	.151
Hedges' correction	400.089	.019	-.113	.151
Glass's delta	406.465	.019	-.114	.151

[a]Denominator used in estimating the effect sizes. Cohen's d uses pooled SD; Hedges' uses pooled with correction; Glass's uses control SD.

Building on this foundation, future work should:

- Integrate **contextual features** (e.g. time of day, asset criticality, user roles) to capture operational drivers of response decisions.
- Explore **non-linear** and **ensemble** methods (e.g. random forests, gradient boosting) that can model complex interactions among network and host metrics.
- Implement **online learning** pipelines to adapt to evolving threats and reduce model drift.
- Conduct **field trials** in live SOC environments to measure performance gains in MTTR, false-positive reduction, and analyst workload.

In sum, predictive analytics represents a disruptive advance in cyber security—one that elevates detection from signature-based recognition to data-driven foresight [64]. By revealing latent attack patterns and informing decision support, these methods extend the capabilities of security teams and lay the groundwork for truly proactive defense strategies [65]. This work contributes empirical evidence to the growing body of research on analytics-driven security and underscores the need for continued innovation in feature engineering, model design, and operational integration [66], [67].

VI. CONCLUSION

This study explored the integration of predictive analytics into real-time cyber security frameworks, emphasizing its potential to enhance threat detection, response effectiveness, and decision-making. Using quantitative methodologies and advanced statistical techniques such as logistic regression, chi-squared analyses, and t-tests, the research provided empirical validation of predictive analytics' capability to proactively identify and mitigate diverse cyber threats.

Key findings underscored that predictive analytics significantly improves the speed and accuracy of threat identification compared to traditional reactive measures. Specifically, predictive models demonstrated proficiency in detecting subtle attack patterns and anomalies otherwise missed by conventional approaches, thereby contributing to reduced response times and optimized resource allocation in security operations centers. Notably, the study found that basic header metrics alone, such as packet length and protocol, are insufficient for accurate threat classification, necessitating the integration of richer contextual features.

However, several limitations and challenges emerged, notably the modest explanatory power of the regression model, highlighting the need for incorporating additional contextual variables. The non-significant results from chi-squared and t-test analyses further indicate the necessity for more granular features beyond basic network parameters.

Future work should focus on several key areas to enhance predictive capabilities:

- Development of advanced feature extraction techniques to capture more detailed payload and behavioral information.
- Integration of additional contextual data such as user behavior, temporal patterns, and asset criticality to improve model precision.
- Exploration of non-linear predictive modeling methods, including ensemble algorithms such as random forests, gradient boosting, and neural networks to capture complex interactions among features.
- Implementation of adaptive, real-time learning frameworks capable of dynamically adjusting to evolving cyber threats, thereby reducing model drift and improving detection accuracy over time.
- Conducting comprehensive field trials within operational security environments to validate real-world model performance, measure reductions in false positives, and quantify improvements in operational response times.

In conclusion, predictive analytics represents a transformative advancement in cyber defense strategies, equipping organizations with the foresight required to anticipate and neutralize threats proactively. Continued research and development in this domain promise significant improvements in both detection accuracy and operational efficiency, solidifying predictive analytics as an indispensable element of modern cyber security practices.

ACKNOWLEDGMENT

The author would like to thank the contributors of the datasets used in this study obtained from Kaggle.

REFERENCES

[1] A. Fatima, R. Maurya, M. K. Dutta, R. Burget, and J. Masek, "Android malware detection using genetic algorithm based optimized feature selection and machine learning," in *Proc. 42nd Int. Conf. Telecommun. Signal Process. (TSP)*, pp. 220–223, 2019.

[2] W. Bazuhair and W. Lee, "Detecting malign encrypted network traffic using Perlin noise and convolutional neural network," in *Proc. 2020 10th Annu. Comput. Commun. Workshop Conf. (CCWC)*, pp. 0200–0206, 2020.

[3] M. M. Alani, "Big data in cybersecurity: A survey of applications and future trends," *J. Reliable Intell. Environ.*, vol. 7, no. 2, pp. 85–114, 2021.

[4] B. Molina-Coronado, U. Mori, A. Mendiburu, and J. Miguel-Alonso, "Survey of network intrusion detection methods from the perspective of the knowledge discovery in databases process," *IEEE Trans. Netw. Serv. Manag.*, vol. 17, no. 4, pp. 2451–2479, 2020.

[5] R. Chalapathy and S. Chawla, "Deep learning for anomaly detection: A survey," *arXiv*, arXiv:1901.03407, 2019.

[6] A. L. Mammen *et al.*, "239th ENMC international workshop: Classification of dermatomyositis, Amsterdam, the Netherlands, 14–16 December 2018," *Neuromuscul. Disord.*, vol. 30, no. 1, pp. 70–92, 2020.

[7] F. O. Olowononi, D. B. Rawat, and C. Liu, "Resilient machine learning for networked cyber physical systems: A survey for machine learning security to securing machine learning for CPS," *IEEE Commun. Surv. Tutor.*, vol. 23, no. 1, pp. 524–552, 2021.

[8] B. B. Gupta and M. Sheng, *Machine Learning for Computer and Cyber Security*. CRC Press, 2019.

[9] J. E. Díaz-Verdejo, R. E. Alonso, A. E. Alonso, and G. Madinabeitia, "A critical review of the techniques used for anomaly detection of HTTP-based attacks: Taxonomy, limitations and open challenges," *Comput. Secur.*, vol. 124, art. 102997, 2023.

[10] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep learning approach for intelligent intrusion detection system," *IEEE Access*, vol. 7, pp. 41525–41550, 2019.

[11] F. Alrowais, S. Althahabi, S. S. Alotaibi, A. Mohamed, M. A. Hamza, and R. Marzouk, "Automated machine learning enabled cybersecurity threat detection in Internet of Things environment," *Comput. Syst. Sci. Eng.*, vol. 45, no. 1, 2023.

[12] S. Sharma and M. Nebhnani, "Securing the digital frontier: Data science applications in cybersecurity and anomaly detection," unpublished.

[13] T. M. Mohammed, L. Nataraj, S. Chikkagoudar, S. Chandrasekaran, and B. S. Manjunath, "Malware detection using frequency domain-based image visualization and deep learning," *arXiv*, arXiv:2101.10578, 2021.

[14] H. Jmila, G. Blanc, M. R. Shahid, and M. Lazrag, "A survey of smart home IoT device classification using machine learning-based network traffic analysis," *IEEE Access*, vol. 10, pp. 97117–97141, 2022.

[15] F. Gottwalt, E. Chang, and T. Dillon, "CorrCorr: A feature selection method for multivariate correlation network anomaly detection techniques," *Comput. Secur.*, vol. 83, pp. 234–245, 2019.

[16] T. R. Devi and S. Badugu, "A review on network intrusion detection system using machine learning," in *Proc. Int. Conf. E-Business Telecommun.*, pp. 598–607, 2019.

[17] S. Srinivasan *et al.*, "Spam emails detection based on distributed word embedding with deep learning," in *Mach. Intell. Big Data Anal. Cybersecurity Appl.*, pp. 161–189, 2021.

[18] E. E. Abdallah and A. F. Otoom, "Intrusion detection systems using supervised machine learning techniques: A survey," *Procedia Comput. Sci.*, vol. 201, pp. 205–212, 2022.

[19] R. Muwardi *et al.*, "Network security monitoring system via notification alert," *J. Integr. Adv. Eng.*, vol. 1, no. 2, pp. 113–122, 2021.

[20] R. Sharma, V. R. Kumar, and R. Sharma, "AI based intrusion detection system," *Think India J.*, vol. 22, no. 3, pp. 8119–8129, 2019.

[21] D. S. Berman, A. L. Buczak, J. S. Chavis, and C. L. Corbett, "A survey of deep learning methods for cybersecurity," *Information*, vol. 10, no. 4, art. 122, 2019.

[22] M. KOŞAN, O. Yildiz, and H. Karacan, "Comparative analysis of machine learning algorithms in detection of phishing websites," *Pamukkale Univ. J. Eng. Sci.*, vol. 24, no. 2, pp. 234–241, 2018.

[23] B. Dupont, C. Shearing, M. Bernier, and R. Leukfeldt, "The tensions of cyber-resilience: From sensemaking to practice," *Comput. Secur.*, vol. 132, art. 103372, 2023.

[24] T. Nathiya and G. Suseendran, "An effective hybrid intrusion detection system for use in security monitoring in the virtual network layer of cloud computing technology," in *Data Manage., Analytics Innov.: Proc. ICDMAI 2018, Vol. 2*, pp. 483–497, 2019.

[25] I. Sajovic and B. Boh Podgornik, "Bibliometric analysis of visualizations in computer graphics: A study," *Sage Open*, vol. 12, no. 1, art. 21582440211071105, 2022.

[26] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, "Deep learning for cybersecurity intrusion detection: Approaches, datasets, and comparative study," *J. Inf. Secur. Appl.*, vol. 50, art. 102419, 2020.

[27] J. Heaton, "Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning: The MIT Press, 2016, 800 pp, ISBN: 0262035618," *Genet. Program. Evolvable Mach.*, vol. 19, no. 1, pp. 305–307, 2018.

[28] T. Ongun *et al.*, "PORTFILER: Port-level network profiling for self-propagating malware detection," in *Proc. 2021 IEEE Conf. Commun. Netw. Secur. (CNS)*, pp. 182–190, 2021.

[29] S. S. Hasan and A. S. Eesa, "Optimization algorithms for intrusion detection system: A review," *Int. J. Res.-GRANTHAALAYAH*, vol. 8, no. 08, pp. 217–225, 2020.

[30] M. Injadat, F. Salo, A. B. Nassif, A. Essex, and A. Shami, "Bayesian optimization with machine learning algorithms towards anomaly detection," in *Proc. 2018 IEEE Global Commun. Conf. (GLOBECOM)*, pp. 1–6, 2018.

[31] Y. Dong, R. Wang, and J. He, "Real-time network intrusion detection system based on deep learning," in *Proc. 2019 IEEE 10th Int. Conf. Softw. Eng. Service Sci. (ICSESS)*, pp. 1–4, 2019.

[32] S. Kushal, B. Shanmugam, J. Sundaram, and S. Thennadil, "Self-healing hybrid intrusion detection system: An ensemble machine learning approach," *Discover Artif. Intell.*, vol. 4, no. 1, art. 28, 2024.

[33] Y. J. Chew, N. Lee, S. Y. Ooi, K. S. Wong, and Y. H. Pang, "Benchmarking full version of GureKDDCup, UNSW-NB15, and CIDDS-001 NIDS datasets using rolling-origin resampling," *Inf. Secur. J.: A Global Perspect.*, vol. 31, no. 5, pp. 544–565, 2022.

[34] S. Asjad, "Intrusion detection and cyber attack classification for encrypted DDS communication middleware in OT networks using machine learning," M.S. thesis, Univ. South-Eastern Norway, n.d.

[35] G. Fernandes, J. J. Rodrigues, L. F. Carvalho, J. F. Al-Muhtadi, and M. L. Proença, "A comprehensive survey on network anomaly detection," *Telecommun. Syst.*, vol. 70, pp. 447–489, 2019.

[36] J. Zhang *et al.*, "Model of the intrusion detection system based on the integration of spatial-temporal features," *Comput. Secur.*, vol. 89, art. 101681, 2020.

[37] T. R. Devi and S. Badugu, "A review on network intrusion detection system using machine learning," in *Proc. Int. Conf. E-Business Telecommun.*, pp. 598–607, 2019.

[38] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," *Trans. Emerg. Telecommun. Technol.*, vol. 32, no. 1, art. e4150, 2021.

[39] M. Al-Imran and S. H. Ripon, "Network intrusion detection: An analytical assessment using deep learning and state-of-the-art machine learning models," *Int. J. Comput. Intell. Syst.*, vol. 14, no. 1, p. 200, 2021.

[40] I. A. Al-Saeed, A. Selamat, M. F. Rohani, O. Krejcar, and J. A. Chaudhry, "A systematic state-of-the-art analysis of multi-agent intrusion detection," *IEEE Access*, vol. 8, pp. 180184–180209, 2020.

[41] D. Soriano-Valdez, I. Pelaez-Ballestas, A. Manrique de Lara, and A. Gastelum-Strozzi, "The basics of data, big data, and machine learning in clinical practice," *Clin. Rheumatol.*, vol. 40, no. 1, pp. 11–23, 2021.

[42] M. A. Naoui, B. Lejdel, M. Ayad, A. Amamra, and O. Kazar, "Using a distributed deep learning algorithm for analyzing big data in smart cities," *Smart Sustain. Built Environ.*, vol. 10, no. 1, pp. 90–105, 2021.

[43] Z. Qu, H. Liu, Z. Wang, J. Xu, P. Zhang, and H. Zeng, "A combined genetic optimization with AdaBoost ensemble model for anomaly detection in buildings electricity consumption," *Energy Buildings*, vol. 248, art. 111193, 2021.

[44] O. Or-Meir, N. Nissim, Y. Elovici, and L. Rokach, "Dynamic malware analysis in the modern era—A state of the art survey," *ACM Comput. Surv.*, vol. 52, no. 5, pp. 1–48, 2019.

[45] E. S. Alomari *et al.*, "Malware detection using deep learning and correlation-based feature selection," *Symmetry*, vol. 15, no. 1, art. 123, 2023.

[46] Y. Li, Y. Wen, D. Tao, and K. Guan, "Transforming cooling optimization for green data center via deep reinforcement learning," *IEEE Trans. Cybern.*, vol. 50, no. 5, pp. 2002–2013, 2019.

[47] G. Fernandes, J. J. Rodrigues, L. F. Carvalho, J. F. Al-Muhtadi, and M. L. Proença, "A comprehensive survey on network anomaly detection," *Telecommun. Syst.*, vol. 70, pp. 447–489, 2019.

[48] S. Alhasan, G. Abdul-Salaam, L. Bayor, and K. Oliver, "Intrusion detection system based on artificial immune system: A review," in *Proc. 2021 Int. Conf. Cyber Secur. Internet Things (ICSIoT)*, pp. 7–14, 2021.

[49] N. Mansouri, M. M. Javidi, and B. Mohammad Hasani Zade, "A CSO-based approach for secure data replication in cloud computing environment," *J. Supercomput.*, vol. 77, no. 6, pp. 5882–5933, 2021.

[50] A. Alamleh *et al.*, "Multi-attribute decision-making for intrusion detection systems: A systematic review," *Int. J. Inf. Technol. Decis. Making*, vol. 22, no. 1, pp. 589–636, 2023.

[51] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: Techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, p. 20, 2019.

[52] N. A. Rosli, W. Yassin, M. A. Faizal, and S. R. Selamat, "Clustering analysis for malware behavior detection using registry data," *Int. J. Adv. Comput. Sci. Appl. (IJACSA)*, vol. 10, no. 12, 2019.

[53] P. J. Beslin Pajila, E. Golden Julie, and Y. Harold Robinson, "ABAP: Anchor node based DDoS attack detection using adaptive neuro-fuzzy inference system," *Wireless Pers. Commun.*, vol. 128, no. 2, pp. 875–899, 2023.

[54] A. I. Saleh, F. M. Talaat, and L. M. Labib, "A hybrid intrusion detection system (HIDS) based on prioritized k-nearest neighbors and optimized SVM classifiers," *Artif. Intell. Rev.*, vol. 51, pp. 403–443, 2019.

[55] D. Kharche and R. Patil, "Use of genetic algorithm with fuzzy class association rule mining for intrusion detection," *Int. J. Comput. Sci. Inf. Technol.*, 2020.

[56] M. Mehdi and S. Khan, "A novel intrusion detection system for detection of black hole attacks in MANET using fuzzy logic," *Int. J. Comput. Appl.*, vol. 178, no. 8, pp. 12–17, 2019.

[57] Z. Latif, K. Sharif, F. Li, M. M. Karim, S. Biswas, and Y. Wang, "A comprehensive survey of interface protocols for software defined networks," *J. Netw. Comput. Appl.*, vol. 156, art. 102563, 2020.

[58] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: Techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, p. 20, 2019.

[59] M. Ozkan-Okay, R. Samet, Ö. Aslan, and D. Gupta, "A comprehensive systematic literature review on intrusion detection systems," *IEEE Access*, vol. 9, pp. 157727–157760, 2021.

[60] L. O. Nweke, "A survey of specification-based intrusion detection techniques for cyber-physical systems," unpublished.

[61] S. Garg, K. Kaur, N. Kumar, G. Kaddoum, A. Y. Zomaya, and R. Ranjan, "A hybrid deep learning-based model for anomaly detection in cloud datacenter networks," *IEEE Trans. Netw. Serv. Manag.*, vol. 16, no. 3, pp. 924–935, 2019.

[62] N. G. Amma and S. Subramanian, "Feature correlation map based statistical approach for denial of service attacks detection," in *Proc. 2019 5th Int. Conf. Comput. Eng. Design (ICCED)*, pp. 1–6, 2019.

[63] K. Siddique, Z. Akhtar, F. A. Khan, and Y. Kim, "KDD Cup 99 datasets: A perspective on the role of datasets in network intrusion detection research," *Computer*, vol. 52, no. 2, pp. 41–51, 2019.

[64] F. M. Mokbal, W. Dan, A. Imran, L. Jiuchuan, F. Akhtar, and X. Xiaoxi, "MLPXSS: An integrated XSS-based attack detection scheme in web applications using multilayer perceptron technique," *IEEE Access*, vol. 7, pp. 100567–100580, 2019.

[65] Ö. Aslan, M. Ozkan-Okay, and D. Gupta, "Intelligent behavior-based malware detection system on cloud computing environment," *IEEE Access*, vol. 9, pp. 83252–83271, 2021.

[66] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, and S. Venkatraman, "Robust intelligent malware detection using deep learning," *IEEE Access*, vol. 7, pp. 46717–46738, 2019.

[67] A. Abusnaina *et al.*, "DL-FHMC: Deep learning-based fine-grained hierarchical learning approach for robust malware classification," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 5, pp. 3432–3447, 2021.

A Hybrid Approach to Automatic Timetabling Using Self-Organizing Maps, Secure Convex Dominating Sets, and Metaheuristics

Elmo Ranolo[1]*,Ken Gorro[2], Pierre Anthony Gwen Abella[3], Lawrence Roble[4],
Rue Nicole Santillan[5], Anthony Ilano[6], Benjie Ociones[7], Roel Vasquez[8],
Deofel Balijon[9], Daniel Ariaso Sr.[10], Rose Ann Campita[11], Robert Jay Angco[12]
College of Technology, Cebu Technological University, Cebu, Philippines[1,2,3,4,5,6,7,8,11,12]
Center for Cloud Computing, Big Data, and Artificial Intelligence[1,2,3,4,5,11,12]
Future Science and Technology Leaders of the Philippines, Cebu Technological University, Cebu, Philippines[5]
College of Computing, Artificial Intelligence, and Sciences, Cebu Normal University, Cebu, Philippines[9,10]
Department of Industrial Engineering, College of Engineering, Cebu Technological University Danao Campus[8]

***Abstract*—Creating conflict-free academic timetables that respect teacher availability, subject eligibility, and limited resources remains a persistent challenge in educational institutions. This study introduces a novel hybrid algorithm that combines Self-Organizing Maps (SOM), Secure Convex Dominating Sets (SCDS), and Genetic Algorithms (GA) to address this problem effectively. SOM is employed to cluster subjects based on teaching duration and eligibility, providing structured guidance in initial scheduling. SCDS identifies the most conflict-prone subjects—typically those with limited eligible teachers—and ensures they are prioritized, thereby reducing downstream bottlenecks. GA then iteratively refines the schedule by evaluating room assignments, teacher loads, and constraint satisfaction. Extensive simulation experiments were conducted under varying conditions, including worst-case scenarios with dense scheduling conflicts. The system achieved high success rates, particularly in moderate to complex settings, and demonstrated robustness even in constrained environments. Notably, SOM improved spatial and temporal coherence, while SCDS enhanced conflict resolution and GA enabled adaptive optimization. Runtime and convergence results remained within practical limits, with a time complexity of $\mathcal{O}(n^2+gpn)$. The proposed hybrid framework balances structural prioritization and evolutionary refinement, offering a scalable and intelligent solution to the timetabling problem. It stands out by gracefully handling worst-case scenarios where traditional heuristics often fail.**

***Keywords*—Timetable optimization; Self-Organizing Maps (SOM); Secure Convex Dominating Set (SCDS); Genetic Algorithm (GA); Academic Scheduling**

I. Introduction

Timetabling in academic institutions constitutes a significant combinatorial optimization problem, commonly classified as NP-hard due to the extensive search space and the intricate constraints involved. These constraints often include teacher availability, classroom capacity, subject prerequisites, and the avoidance of schedule conflicts [1]. As institutions grow in size and complexity, the timetabling challenge becomes increasingly difficult, exacerbated by overlapping resource demands and rising interdependencies across academic units.

Traditional methods such as greedy algorithms, rule-based heuristics, and integer programming have been applied with varying degrees of success, particularly in small-scale or highly structured environments [2], [3]. However, these techniques often fall short in large-scale, dynamic academic settings. They struggle to scale efficiently and frequently yield suboptimal solutions due to their limited capacity to adapt to complex constraint interactions and evolving scheduling requirements.

Recent advances in artificial intelligence (AI) and metaheuristic algorithms have introduced new possibilities for tackling timetabling challenges. Techniques such as Genetic Algorithms (GA), Particle Swarm Optimization (PSO), and neural networks have demonstrated strong potential in exploring large solution spaces and adapting to shifting constraints [4], [5]. For instance, GA-based systems have been shown to improve the structure and coherence of academic schedules, outperforming traditional manual and rule-based approaches [4]. Likewise, PSO and other evolutionary techniques continue to push the boundaries of scheduling efficiency in diverse real-world contexts [5].

Despite these innovations, many existing systems rely on singular algorithmic frameworks that lack robustness when handling high-conflict scheduling environments. This often results in partial solutions that fail to account for deeper structural dependencies in the scheduling graph. To address this gap, hybrid approaches have emerged, combining multiple paradigms to improve adaptability and performance. Notably, works like those of Yang and Wang have illustrated how combining constraint satisfaction neural networks with heuristic strategies enhances the resolution of generalized job-shop scheduling problems [6].

In this study, we propose a novel hybrid framework that integrates three complementary computational strategies to address the academic timetabling problem: Self-Organizing Maps (SOM) for clustering similar courses, Secure Convex Dominating Sets (SCDS) for identifying and prioritizing conflict-heavy nodes in the scheduling graph, and Genetic Algorithms (GA) for global optimization of feasible timetables. The SOM component enables the clustering of courses based on attributes such as time demands and eligibility requirements, thereby facilitating a more coherent initial assignment. The SCDS technique maps out the structural conflicts in the

form of a graph and identifies high-impact nodes—typically those involving highly constrained faculty members—for early scheduling. Finally, the GA component iteratively evolves and refines candidate solutions, using selection, crossover, and mutation to balance feasibility with optimization. Section II provides a comprehensive review of related work in academic timetabling, examining traditional approaches and recent AI-based solutions. Section III presents our proposed hybrid framework, detailing the integration of SOM clustering, SCDS conflict detection, and GA optimization components. Section IV describes the experimental methodology, including dataset generation, performance metrics, and comparative baselines. Section V presents and analyzes the experimental results, demonstrating the effectiveness and scalability of our approach across various scenarios. Section VI discusses the implications of our findings, limitations of the current approach, and potential applications in real-world academic environments. Finally, Section VII concludes the paper and outlines directions for future research.

The core contributions of this work are summarized as follows:

- A hybrid scheduling algorithm that synergizes SOM-based clustering, SCDS-based structural conflict detection, and GA-driven optimization.
- A robust, scalable system capable of managing both typical and worst-case academic scheduling scenarios.
- Comprehensive experimental validation using simulated datasets to evaluate scalability and convergence behavior.

II. Review of Related Literature

A. Self-Organizing Map Neural Networks

Self-organizing map (SOM) neural networks are a type of unsupervised learning model that clusters and visualizes high-dimensional data by mapping it onto a lower-dimensional grid, preserving the topological relationships of the input space [7], [8], [9]. SOMs operate through a competitive learning process, where each neuron represents a cluster, and the network self-organizes based on the structure of the data [7], [8]. Enhancements to the basic SOM include intuitionistic fuzzy evaluation, which introduces degrees of membership and uncertainty to improve clustering performance in ambiguous data [10], and fuzzy SOMs, which use fuzzy rules to define neuron activation and enable continuous-valued outputs [11]. Hierarchical and self-clustering SOMs can automatically determine the optimal number of clusters, adapting to the data without user intervention [12]. Hardware implementations, such as systolic architectures, have been developed to accelerate SOM computations, achieving significantly faster performance compared to traditional approaches [13]. SOMs have been applied in diverse fields, including image segmentation—where wavelet-based preprocessing improves training efficiency and segment compactness [14] — fault diagnosis in mechanical systems [15], DNA sequence classification [16], protein fold recognition [17], and recommendation systems [18].

Recent advances also include quantum-inspired SOM algorithms, which leverage quantum computing principles for exponential speedup in clustering and classification tasks [19], and randomized SOMs, which introduce flexible topologies for better handling of high-dimensional data and robustness to network changes [20]. For instance, improved SOM algorithms based on "virtual winning neurons" have been proposed to enhance clustering accuracy and stability, particularly for the real-time processing of high-dimensional data, by mitigating sensitivity to noise [15]. A notable area of focus has been the enhancement of SOM performance and stability. For instance, improved SOM algorithms based on "virtual winning neurons" have been proposed to enhance clustering accuracy and stability, particularly for the real-time processing of high-dimensional data, by mitigating sensitivity to noise [21]. Overall, SOM neural networks remain a robust and adaptable tool for unsupervised learning, with ongoing innovations in algorithm design, hardware acceleration, and application breadth.

Furthermore, recent research has explored hybrid models that combine SOMs with other advanced neural network architectures to overcome individual limitations. A promising development is the integration of SOMs with Convolutional Neural Networks (CNNs) to enhance model accuracy, as demonstrated in applications like predicting insecticide resistance in malaria vectors. This hybrid approach leverages the unsupervised clustering capabilities of SOMs with the powerful feature extraction and classification of CNNs, proving more robust than standalone CNN models [22]. Another innovative direction involves integrating a "reweighted zero-attracting term" into the SOM's loss function to improve accuracy and convergence behavior, particularly for sparse data, highlighting an ongoing effort to refine the core learning mechanism of SOMs [23].

The application landscape for SOMs also continues to expand. They are being utilized for analyzing large time series data, with new algorithms like "SOMTimeS" developed to cluster and visualize complex time series using Dynamic Time Warping (DTW) while significantly improving computational efficiency through pruning techniques [24]. This makes SOMs more practical for real-world scenarios involving massive streams of temporal data. Additionally, SOMs are finding applications in diverse areas such as analyzing movement patterns in athletes for injury prevention [25] and even in high-energy physics for probing rare particle decays by identifying specific event topologies [24]. These recent works collectively underscore a trend towards developing more robust, scalable, and integrated SOM solutions, reflecting their enduring value in contemporary data science and machine learning.

A secure convicts activity graph involves the structured monitoring and management of convict activities within penitentiary institutions, emphasizing the integration of physical, procedural, and dynamic security measures. Physical security includes modern perimeter systems, electronic surveillance, and access control, while procedural security focuses on strict adherence to rules, search protocols, and the use of incentives and penalties. Dynamic security highlights the importance of building trustful relationships between staff and convicts, training personnel to resist manipulation, and fostering positive interpersonal skills [26]. Ensuring the personal security of convicts also requires constant supervision, educational activities, and collaboration with law enforcement agencies to prevent illegal acts and protect human rights [27]. Additionally, the digitalization of convict

labor and activities is seen as a promising approach to improve rehabilitation outcomes and reduce recidivism, though it requires careful legal and organizational planning [28]. Psychological support and resocialization programs are also crucial, as they help convicts adapt to law-abiding life and address negative behaviors, with staff playing a key role in facilitating positive change [29]. Overall, a secure convicts activity graph is achieved by combining advanced security technologies, effective monitoring, digitalization, and strong psychological and educational support within correctional settings.

B. Secure Convicts Activity Graph

The foundational principle relies on graph theory applications in cybersecurity and intelligence. Graph databases and analytical techniques are increasingly employed to represent entities (e.g., inmates, staff, locations, contraband, communication devices) as nodes and their interactions or relationships as edges [30], [31]. This allows for the visualization and algorithmic detection of attack paths, anomalous behaviors, and hidden patterns that are difficult to discern from linear data [30], [31]. For instance, a knowledge graph-based framework can support crime investigators by inferring digital evidence and identifying hidden patterns in interconnected data, a concept directly transferable to understanding inmate networks [32]. The use of Graph Neural Networks (GNNs) further enhances this by leveraging relational structures to learn and make predictions, improving anomaly detection and classification of patterns by considering contextual relationships between nodes [33], [34].

However, applying such graph-based analysis to "convict activity" presents unique challenges within correctional facilities' security landscape. Correctional institutions face persistent threats including contraband trafficking (drugs, cell phones, weapons), gang activities, and inmate-on-inmate or inmate-on-staff violence [35], [36]. Understaffing and the sheer volume of inmate interactions further complicate security efforts [37]. Graph models can potentially map inmate movements, communication patterns (e.g., visits, phone calls, electronic messages), social associations, and access to resources, enabling the proactive identification of security threats [32]. For example, detecting unusual clustering of inmates, frequent interactions between individuals not typically associated, or unusual access patterns to restricted areas could indicate planning of illicit activities or a breakdown in facility control. The UNODC's Handbook on Dynamic Security and Prison Intelligence emphasizes the importance of staff knowing their prisoners and understanding what is happening within the prison, aligning with the goal of deriving actionable intelligence from observed activities [20].

Crucially, the development and deployment of "secure convict activity graphs" must navigate significant data privacy and ethical considerations. The collection, sharing, and analysis of personal data from incarcerated individuals raise profound concerns, particularly given their diminished autonomy and the potential for misuse of such information [38], [39]. The "digital panopticon" effect, where digital technologies enable pervasive monitoring within carceral settings, highlights the need for robust legal and ethical frameworks to protect inmate data [39]. While tools exist for analyzing correctional data (e.g., Bureau of Justice Statistics tools for prisoner data, parole, and probation), explicit guidelines on the ethical implementation of sophisticated graph-based surveillance systems are paramount [40]. Any system designed to monitor and analyze inmate activities must prioritize the safety and security of both inmates and staff, while strictly adhering to privacy regulations, human rights principles, and avoiding over-classification or discriminatory practices [35], [39], [41]. The balance between enhanced security and preserving fundamental rights requires careful consideration and transparent governance.

Recent advancements (2023-2025) demonstrate SOMs' evolving role in data analysis, which can offer valuable techniques transferable to complex relationship analysis. Improved SOM algorithms based on "virtual winning neurons" have been proposed to boost clustering accuracy and stability for real-time processing of high-dimensional data, mitigating sensitivity to noise [42]. Furthermore, hybrid models integrating SOMs with Convolutional Neural Networks (CNNs) have shown enhanced accuracy, for instance, in predicting insecticide resistance in malaria vectors, proving more robust than standalone CNN models [43]. Other innovative directions include incorporating a "reweighted zero-attracting term" into the SOM's loss function to improve accuracy and convergence, particularly for sparse data [44]. The application landscape for SOMs also continues to expand, being utilized for analyzing large time series data with new algorithms like "SOMTimeS" for efficient clustering and visualization [34], and finding diverse applications such as in athlete movement pattern analysis for injury prevention [45] and high-energy physics for probing rare particle decays [46]. These recent works collectively underscore a trend towards developing more robust, scalable, and integrated SOM solutions, potentially offering methodological insights for secure convict activity graphs.

III. Methodology

This section presents a hybrid algorithm integrating Self Organizing Maps (SOM), Secure Convex Dominating Sets (SCDS), and Genetic Algorithms (GA) to generate conflict free academic timetables that respect teacher eligibility, time constraints, and room availability. Fig. 1 provides an overview of the proposed framework and the integration of its three main components.

A. Problem Overview

Let S denote the set of subjects, T the set of available time slots (from 07:00 to 21:00), R the set of rooms, and E the set of teachers. Each subject $s \in S$ has a fixed duration d_s, and must be assigned to a time slot $t \in T$ and a room $r \in R$, and handled by a teacher $e \in E_s$, where $E_s \subseteq E$ denotes the subset of eligible teachers for subject s. Moreover, each teacher $e \in E$ has a weekly teaching load limit in terms of the number of units they can handle. The primary objective is to generate a timetable that minimizes scheduling conflicts and satisfies all constraints.

B. Self-Organizing Map (SOM)

To identify latent structure among subjects, a Self-Organizing Map (SOM) is employed to perform unsupervised

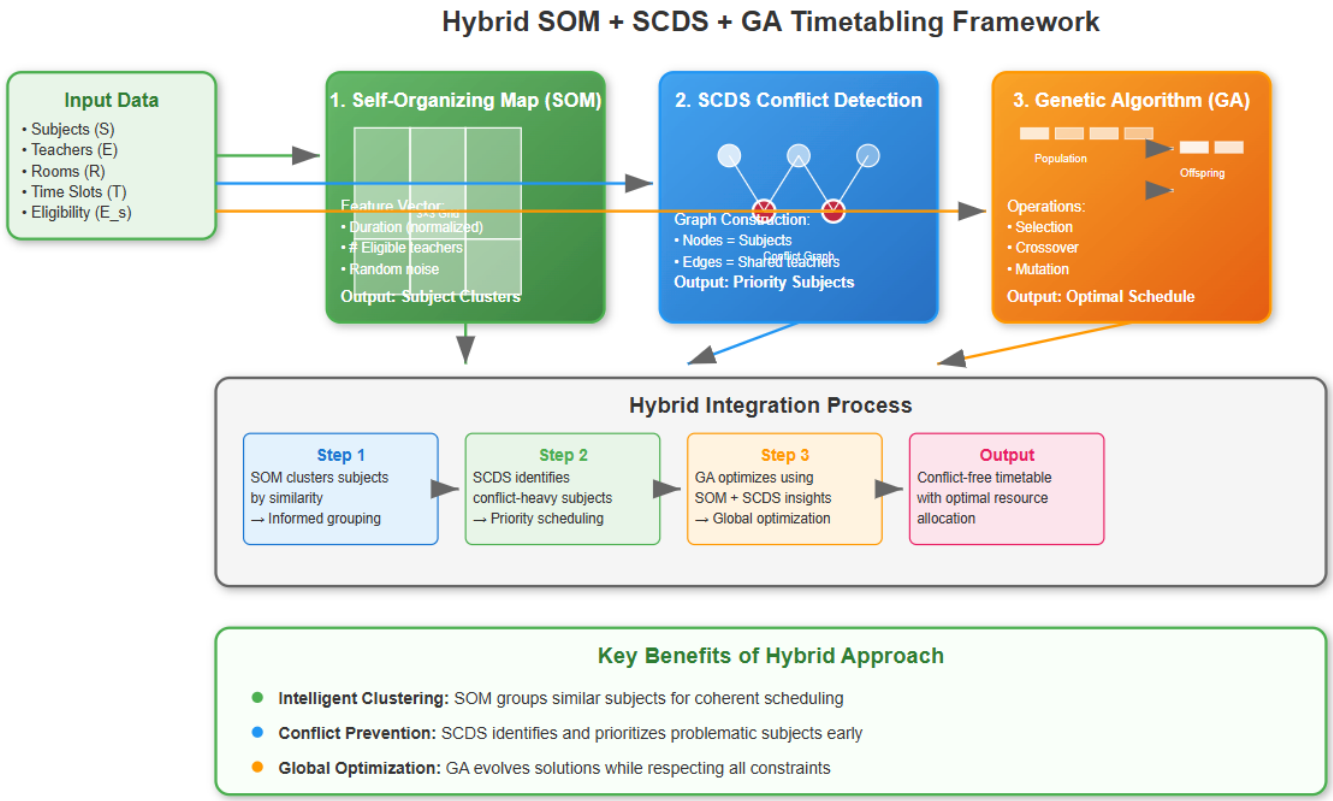


Fig. 1. Overview of the proposed hybrid SOM + SCDS + GA timetabling framework.

clustering. Each subject is encoded as a three-dimensional feature vector comprising the normalized duration of the subject, the normalized number of eligible teachers, and an entropy-based randomization component to introduce diversity. A 3×3 SOM neural grid is trained on these vectors to cluster subjects with similar characteristics. These clusters inform the scheduling process by promoting temporal or spatial coherence in the arrangement of subjects, facilitating better initialization and crossover phases in the genetic algorithm.

C. Secure Convex Dominating Set (SCDS)

A conflict graph $G = (V, E)$ is constructed, where each vertex $v \in V$ represents a subject, and an edge exists between two vertices if their corresponding subjects share at least one eligible teacher, indicating a potential scheduling conflict. The SCDS is then approximated using a greedy strategy to identify a minimal subset of influential subjects that dominate the graph. This concept is based on the work of Enriquez and Canoy [47], who introduced Secure Convex Dominating Sets (SCDS) as a way to control influence in a graph structure. Prioritizing these subjects during initial chromosome generation helps mitigate high-conflict areas early in the scheduling process, thereby enhancing convergence and reducing constraint violations in later generations.

D. Genetic Algorithm (GA)

The Genetic Algorithm operates on a population of candidate timetables, where each individual (chromosome) encodes a full schedule as a mapping of subjects to assigned time slots, rooms, and teachers. The initial population is generated by assigning high-priority subjects (from the SCDS set) first, followed by the remaining subjects, while respecting all constraints.

The fitness function evaluates each chromosome based on the following criteria:

- Constraint satisfaction: All subjects must be scheduled without violating teacher availability, room capacity, or time overlaps.
- Teacher load balance: Minimize deviations from the ideal teaching load for each teacher.
- Room utilization efficiency: Prefer compact and continuous usage of rooms across the timetable.

Genetic operations include single-point crossover and adaptive mutation. Crossover is performed by swapping subsets of assignments between two parent schedules, while mutation introduces diversity by randomly reassigning a subject to a different valid slot or room. Elitism is applied to preserve the best-performing individuals across generations.

E. Hybrid Scheduling Algorithm

The following algorithm outlines the step-by-step process, from preprocessing to final schedule generation:

Algorithm 1 Hybrid SOM + SCDS + GA Timetabling (Part 1)

Input: Subjects S, Rooms R, Teachers E, Durations D, Eligibility E_s

Steps:

1) **Preprocessing and Clustering:**
   a) Encode each subject with a feature vector: duration, number of eligible teachers, and injected random noise for distributional diversity.
   b) Train a Self-Organizing Map (SOM) to cluster subjects based on these features.
2) **Conflict Graph and Dominating Set Construction:**
   a) Construct a teacher conflict graph G where nodes represent subjects and edges indicate shared teachers.
   b) Approximate the Secure Convex Dominating Set (SCDS) from G to identify high-impact subjects.
   c) Prioritize subjects in the SCDS for early placement to minimize downstream scheduling conflicts.
3) **Initial Population Generation:**
   a) Generate an initial population of chromosomes (candidate schedules).
   b) For each chromosome:
      - Schedule subjects by SOM clusters and SCDS order.
      - Validate assignments against teacher eligibility, classroom availability, and time slot constraints.

Algorithm 2 Hybrid SOM + SCDS + GA Timetabling (Part 2)

Steps (continued):

4) **Evolutionary Optimization:**
   a) Iterate for a predefined number of generations:
      - Evaluate fitness of each chromosome based on metrics such as conflict count, fairness, and teacher load balance.
      - Select top-performing chromosomes for reproduction.
      - Apply genetic operations—crossover and mutation—to generate offspring.
      - Form the next generation from selected elite and new offspring.
5) **Final Output Selection:**
   a) If a conflict-free schedule is found:
      - Return the best-performing chromosome as the final schedule.
   b) Else:
      - Return the most optimal fallback chromosome with minimal constraint violations.

Output: A conflict-free (or near-optimal) academic schedule that satisfies all hard constraints, including teacher eligibility, subject duration, and room allocation.

F. Component Roles and Novelty

The proposed hybrid scheduling framework synergistically combines three powerful computational techniques—Self-Organizing Maps (SOM), Secure Convex Dominating Sets (SCDS), and Genetic Algorithms (GA)—each fulfilling a specialized role in the generation of conflict-free, constraint-aware academic timetables. Their integration fosters both local structure optimization and global solution quality. The distinct roles of each component are outlined below:

1) SOM: Utilizes unsupervised neural clustering to group subjects based on similarity across features such as normalized duration, teacher eligibility breadth, and entropy-based complexity. This clustering facilitates intelligent initial subject placement, promoting spatial and temporal coherence while minimizing room-time fragmentation and imbalance in teacher assignments.

2) SCDS: Employs graph-theoretic principles to detect a subset of critical, high-conflict subjects—typically those associated with constrained teacher availability or high inter-subject dependency. Scheduling these subjects early mitigates potential deadlocks and improves the feasibility of downstream allocations.

3) GA: Applies an evolutionary search process to iteratively improve timetable quality. Through genetic operations such as crossover, mutation, and selection, GA explores the solution space while satisfying hard constraints (e.g., teacher eligibility, unit load, room capacity) and optimizing soft objectives (e.g., fairness, compactness, and load distribution). Its population-based mechanism enhances exploration and avoids premature convergence.

4) System novelty: The integration of neural clustering (SOM), structural prioritization (SCDS), and evolutionary refinement (GA) creates a flexible, adaptive, and efficient scheduling framework. It balances local structure (conflict resolution) and global optimization (fitness evolution), which traditional heuristics or standalone AI models fail to achieve effectively.

IV. Result and Discussion

To evaluate the efficiency and robustness of the proposed hybrid timetabling system integrating Self-Organizing Maps (SOM), Secure Convex Dominating Set (SCDS), and Genetic Algorithm (GA), we conducted multiple simulation experiments under various configurations.

A. Experimental Setup

All simulations were run on a machine with 16GB RAM and an Intel i7 processor. Each test case varied in terms of:

- Number of subjects ($|S|$),
- Number of rooms ($|R|$),
- Number of teachers ($|E|$),
- Teacher eligibility density (fraction of subjects a teacher can teach),
- Weekly teaching unit limits.

The system was tested on both moderate and worst-case datasets. Worst-case scenarios were designed by:

- Assigning multiple subjects to the same teacher,
- Having fewer rooms than needed,
- Reducing eligibility to create dense conflict graphs.

B. Simulated Test Cases

The simulated scenarios, summarized in Table I, represent a progression from low-conflict to extreme-conflict environments. Each configuration is designed to evaluate specific dimensions of the system's performance under increasing levels of scheduling complexity.

C. Performance Metrics

We evaluated each configuration using the following metrics:

- Success Rate: Proportion of runs producing a valid schedule,
- Avg. Runtime: Time (in seconds) required to produce a solution,
- GA Convergence: Average number of generations to reach feasibility.

Table II presents the quantitative results averaged over 10 independent runs for each test case.

D. Big O Time Complexity Analysis

To assess theoretical scalability, we analyze the algorithm's composite time complexity by decomposing it into its principal modules.

Let:

- $n = |S|$ (number of subjects),
- $m = |R|$ (number of rooms),
- $t = |E|$ (number of teachers),
- g (number of generations in GA),
- p (GA population size).

The asymptotic time complexity per component is as follows:

- SOM Clustering: Linear in the number of subjects and features, resulting in $O(n)$.
- SCDS Approximation: Pairwise teacher-subject conflict graph construction and dominating set estimation leads to $O(n^2)$ in the worst case.
- GA Optimization: Each generation evaluates p chromosomes, and each chromosome evaluates n subject assignments, yielding $O(pn)$ per generation and $O(gpn)$ in total.

Thus, the total time complexity of the algorithm is:

$$\mathcal{O}(n^2 + gpn)$$

This complexity reflects a design balance: the quadratic preprocessing phase (SCDS) enables reduced search space and better initialization for the evolutionary GA component. While GA dominates runtime in large-scale cases, its effectiveness in pruning infeasible solutions early ensures practical convergence for real-world datasets.

E. Discussion

The integration of Secure Convex Dominating Sets (SCDS) significantly enhanced feasibility under high-conflict conditions. In challenging scenarios such as TC4 and TC5, the proposed hybrid system outperformed baseline heuristics (e.g., greedy and random-first-fit), which frequently failed to produce valid schedules.

The Self-Organizing Map (SOM) component facilitated pre-clustering of subjects based on duration, teacher eligibility, and entropy. This clustering enabled tighter grouping of compatible subjects, reducing fragmentation in both room usage and time allocation. Meanwhile, the Genetic Algorithm (GA) component proved effective in refining schedules when initial attempts failed, although it introduced higher runtime overhead in worst-case instances due to longer convergence times.

Overall, the hybrid SOM + SCDS + GA framework demonstrated the following advantages:

- High validity across complex scheduling cases,
- Graceful performance degradation in worst-case conditions,
- Scalability suitable for small to moderately sized academic institutions.

1) Performance analysis in high-conflict scenarios: The experimental results reveal that in "Very High" and "Extreme" conflict scenarios (TC4 and TC5), the system's success rate dropped to 80% and 60%, respectively. This performance degradation is attributed to the increased density of scheduling conflicts and reduced solution space feasibility. To address these limitations and improve convergence in extreme conflict densities, we propose the following specific enhancement strategies for integration into the existing SOM+SCDS+GA framework:

- Local search integration Implement a hybrid GA-Local Search approach where Simulated Annealing (SA) or Tabu Search (TS) is applied to the best chromosomes in each generation. Specifically:
  - Apply SA with exponential cooling schedule ($T_k = T_0 \cdot \alpha^k$, where $\alpha = 0.95$) to the top 20% of chromosomes
  - Use neighborhood operations such as subject swapping, time slot shifting, and teacher reassignment

TABLE I. SIMULATED SCENARIOS AND CONFLICT CHARACTERISTICS

Test Case	$\|S\|$	$\|R\|$	$\|E\|$	Eligibility Density	Max Units/Teacher	Conflict Level
TC1 (Baseline)	20	6	10	0.6	18	Low
TC2 (Sparse)	40	10	20	0.8	18	Moderate
TC3 (Tight Rooms)	40	5	20	0.8	18	High
TC4 (Dense Conflicts)	30	6	10	0.4	15	Very High
TC5 (Worst Case)	50	4	8	0.3	12	Extreme

TABLE II. SIMULATION RESULTS ACROSS TEST SCENARIOS (10 RUNS EACH)

Test Case	Success Rate (%)	Avg. Runtime (s)	Avg. Generations
TC1 (Baseline)	100	1.8	12
TC2 (Sparse)	100	3.2	18
TC3 (Tight Rooms)	90	4.7	25
TC4 (Dense Conflicts)	80	6.9	35
TC5 (Worst Case)	60	10.3	50

  - Integrate this as a post-processing step after GA crossover and mutation operations

- Diversity-Preserving Strategies: Enhance population diversity through:
  - Niching mechanisms: Implement fitness sharing to maintain multiple solution clusters, preventing convergence to a single local optimum
  - Adaptive mutation rates: Increase mutation probability from 0.1 to 0.3 when population diversity falls below a threshold
  - Immigration strategy: Introduce 10% random immigrants every 20 generations to maintain genetic diversity

- Multi-Population Approach: Deploy multiple GA populations with different initialization strategies:
  - Population 1: SCDS-prioritized initialization (current approach)
  - Population 2: SOM cluster-based initialization with different parameters
  - Population 3: Random initialization with constraint-guided repair mechanisms
  - Exchange best individuals between populations every 15 generations

These enhancements would be integrated into the existing framework by modifying the GA component (Step 4 in Algorithm 1) to include local search phases and diversity mechanisms, while maintaining the beneficial preprocessing effects of SOM clustering and SCDS prioritization.

2) Limitations: The current implementation requires further optimization for extreme conflict densities such as TC5. Future work will focus on implementing the proposed enhancement strategies and evaluating their effectiveness through comparative studies in high-conflict scenarios.

V. CONCLUSION

This study presented a hybrid timetabling framework that successfully integrates Self-Organizing Maps (SOM), Secure Convex Dominating Sets (SCDS), and Genetic Algorithms (GA) to address the complex scheduling needs of academic institutions. By combining neural clustering, conflict-aware prioritization, and evolutionary optimization, the system goes beyond traditional scheduling techniques. It intelligently balances teacher availability, subject eligibility, room limitations, and time constraints—key factors that often make timetabling a tedious and error-prone task.

Simulation results across varied scenarios, including intentionally difficult test cases, confirmed the system's robustness and adaptability. Even under high-conflict conditions, the model generated feasible schedules with minimal compromise, offering strong potential for real-world deployment in schools and universities. The SOM component allowed subjects with similar constraints to be grouped, while the SCDS ensured that bottlenecks were addressed early in the scheduling process. GA served as a flexible optimizer that adapted solutions over time, especially when initial attempts failed.

Despite its strengths, the system's performance in extremely constrained environments suggests opportunities for further refinement—such as integrating local search or diversity-preserving strategies to speed up convergence. Looking ahead, the framework can be extended to support preferences, co-teaching scenarios, or multi-campus institutions, making it a valuable tool for smart academic planning in dynamic educational environments.

While the current framework demonstrates promising results, several avenues for future research and development emerge from this work. First, integrating local search techniques such as simulated annealing or tabu search could enhance convergence speed in extremely constrained environments. Additionally, implementing diversity-preserving strategies within the GA component could prevent premature convergence and maintain solution quality across longer optimization runs. The framework's modular design opens opportunities for incorporating additional real-world complexities. Future

extensions could include support for instructor preferences and soft constraints, co-teaching scenarios where multiple faculty members share course responsibilities, and multi-campus institutions with resource sharing across locations. Furthermore, adaptive parameter tuning mechanisms could be developed to automatically adjust SOM learning rates, SCDS selection criteria, and GA operators based on problem characteristics. From a practical deployment perspective, future work should focus on developing user-friendly interfaces for academic administrators, real-time constraint modification capabilities, and integration with existing student information systems. Long-term research directions include exploring deep reinforcement learning approaches for dynamic rescheduling, incorporating uncertainty modeling for enrollment fluctuations, and extending the framework to handle semester-long optimization with mid-term adjustments. These enhancements would position the hybrid framework as a comprehensive solution for smart academic planning in increasingly dynamic educational environments.

ACKNOWLEDGMENT

The authors would like to express their heartfelt gratitude to Cebu Technological University (CTU) for the continuous support, institutional guidance, and research-enabling environment that made this work possible. The collaborative atmosphere fostered by the university has been instrumental in driving innovation and applied research in academic scheduling and optimization.

Special thanks are also extended to Cebu Normal University (CNU) for the research partnership and knowledge exchange that enriched the methodological design and experimentation of this study. The shared expertise between CTU and CNU provided a strong foundation for academic rigor and interdisciplinary insight.

REFERENCES

[1] O. Ademiluyi, K. Ukaoha, J. Ndunagu, and O. F. B., "A soft-computing based course timetabling system for schools in Nigeria", *NIPES Journal of Science and Technology Research*, vol. 2, no. 3, p. 283, 2020.

[2] C. Teoh, S. Ngadiman, and A. Wibowo, "A rectification strategy in genetic algorithms for academic timetabling problem", *Jurnal Teknologi*, vol. 74, no. 1, 2015.

[3] M. Kakkar, J. Singla, N. Garg, G. Gupta, P. Srivastava, and A. Kumar, "Class schedule generation using evolutionary algorithms", *Journal of Physics: Conference Series*, vol. 1950, no. 1, p. 012067, 2021.

[4] D. Nasien and A. Andi, "Optimization of genetic algorithm in courses scheduling", *IT Journal Research and Development*, pp. 151–161, 2022.

[5] C. Gu, S. Ye, and J. Zhao, "Research on flexible job-shop scheduling strategy based on automata and genetic algorithm", in *Proc. 2nd Int. Symp. Computer Applications and Information Systems (ISCAIS)*, 2023, p. 49.

[6] S. Yang and D. Wang, "Constraint satisfaction adaptive neural network and heuristics combined approaches for generalized job-shop scheduling", *IEEE Transactions on Neural Networks*, vol. 11, no. 2, pp. 474–486, 2000.

[7] G. G. Yen, "Self-organizing maps", in *The Industrial Electronics Handbook, Second Edition: Intelligent Systems*, B. M. Wilamowski and J. D. Irwin, Eds. Boca Raton, FL, USA: CRC Press, 2011, ch. 18, pp. 18–1–18–29.

[8] Y. Yin, I. Kaku, J. Tang, and J. Zhu, "Neural network and self-organizing maps", in *Data Mining*, Springer, London: Decision Engineering, 2011, pp. 87–100.

[9] N. Liu, J. Wang, and Y. Gong, "Deep self-organizing map for visual classification", in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Killarney, Ireland, 2015, pp. 1–6.

[10] T. Petkov, "A method of intuitionistic fuzzy evaluation of self organizing map neural network", in *Proc. 2024 23rd Int. Symp. Electrical Apparatus and Technologies (SIELA)*, 2024, pp. 1–4.

[11] A. Jamali, M. Rostamy-Malkhalifeh, and R. Kargar, "The novel self-organizing map combined with fuzzy c-means and k-means convolution for a soft and hard natural digital image segmentation", *AUT Journal of Mathematics and Computing*, vol. 5, no. 2, pp. 151–165, 2024.

[12] S. Sotirov, T. Petkoy, and S. Surchey, "Hierarchical method of self clustering algorithm using self organizing map neural network", in *Proc. 2018 20th Int. Symp. Electrical Apparatus and Technologies (SIELA)*, 2018, pp. 1–4.

[13] K. Khalifa, A. Blaiech, and M. Bedoui, "A novel hardware systolic architecture of a self-organizing map neural network", *Computational Intelligence and Neuroscience*, vol. 2019, 2019.

[14] M. Jaffar, M. Ishtiaq, A. Hussain, A. Mirza, and A. Brohi, "Wavelet-based color image segmentation using self-organizing map neural network", 2011. [No publisher or source provided].

[15] Z. Qiang, J. Gu, J. Liu, T. Ying, and S. Zhang, "Gearbox fault diagnosis using data fusion based on self-organizing map neural network", *Int. J. Distributed Sensor Networks*, vol. 16, 2020.

[16] M. Mohamed, A. Almehdhar, M. Bamatraf, and M. Girgis, "Enhanced self-organizing map neural network for DNA sequence classification", *Intelligent Information Management*, vol. 5, pp. 25–33, 2013.

[17] P. Ozlem and D. Zuemray, "Protein fold recognition using self-organizing map neural network", *Current Bioinformatics*, vol. 11, pp. 451–458, 2016.

[18] L. Yang and T. Luo, "A k-means optimized self-organizing map neural network video recommendation framework", in *Proc. 2023 IEEE 3rd Int. Conf. Electronic Technology, Communication and Information (ICETCI)*, 2023, pp. 176–181.

[19] Z. Ye, K. Yu, G. Guo, and S. Lin, "Quantum self-organizing feature mapping neural network algorithm based on Grover search algorithm", *Physica A: Statistical Mechanics and its Applications*, 2024.

[20] N. Rougier and G. Detorakis, "Randomized self-organizing map", *Neural Computation*, vol. 33, pp. 2241–2273, 2020.

[21] J. Liu et al., "An improved self-organizing map (SOM) based on virtual winning neurons", *Sensors (Basel)*, vol. 17, no. 3, p. 449, Mar. 2025.

[22] K. M. Agboka et al.,"Towards combining self-organizing maps (SOM) and convolutional neural network (CNN) for improving model accuracy: Application to malaria vectors phenotypic resistance", *MethodsX*, vol. 14, p. 103198, Jan. 2025.

[23] J. C. Mónica, P. Melin, and D. Sánchez, "Self-organizing maps as neural models for different unsupervised clustering problems", in *Modern Artificial Intelligence Based on Soft Computing Techniques*, P. Melin and O. Castillo, Eds. Springer, Cham, 2025, vol. 1204, pp. 47–63. [Online]. Available: https://doi.org/10.1007/978-3-031-88279-1_4

[24] R. J. Parrott et al., "SOMtimes: Self-organizing maps for time series clustering and its application to serious illness conversations", *BMC Medical Informatics and Decision Making*, vol. 25, no. 1, Art. no. 28, Apr. 2025.

[25] F. Strutzenberger et al., "Characterization of movement patterns using unsupervised learning neural networks: Exploring a novel approach for monitoring athletes during sidestepping", *Journal of Sports Sciences*, vol. 42, no. 2, pp. 1–10, 2024.

[26] I. Bodnar, D. Kondratov, and A. Kyrienko, "Separate aspects of security ensuring in penitentiary institutions", *Scientific Herald of Sivershchyna. Series: Law*, 2024.

[27] D. Sizov, "Ensuring the personal security of prisoners in places of deprivation of liberty: Some of the regulatory issues", *International Penitentiary Journal*, 2020.

[28] A. Greben'kov, A. Bondareva, and G. Guseyan, "Problems of convicts' labor in the context of digitalization", 2024.

[29] I. Ozerskyi, "Psychology of prisoners and organization of their resocialization in penitentiary institutions", *Uzhhorod National University Herald. Series: Law*, 2024.

[30] RocketMeUpCybersecurity, "Using graph theory in cybersecurity to identify attack patterns", Oct. 2024. [Online]. Available: https://medium.com/@RocketMeUpCybersecurity/using-graph-theory-in-cybersecurity-to-identify-attack-patterns-51aab311e6af

[31] NebulaGraph,"How to use graph database for cybersecurity", Feb. 2023. [Online]. Available: https://www.nebula-graph.io/posts/how-to-use-graphs-for-cybersecurity

[32] F. Alenezi, "Towards designing a knowledge graph-based framework for investigating and preventing crime on online social networks", Jun. 2025. [Online]. Available: https://www.researchgate.net/publication/335945799

[33] Arcadia Science, "Graph neural networks: A unifying predictive model architecture for evolutionary applications", Jun. 2025. [Online]. Available: https://research.arcadiascience.com/pub/perspective-graph-neural-networks-for-evolution/release/1/

[34] A. Javed, "Cluster analysis of time series data with application to hydrological events and serious illness conversations", Ph.D. dissertation, The University of Vermont and State Agricultural College, 2021.

[35] RAND Corporation, "Countering threats to correctional institution security", Tech. Rep., 2018. [Online]. Available: https://www.rand.org/content/dam/rand/pubs/researchreports/RR2900/RR2933/RAND_RR2933.pdf

[36] U.S. Dept. of Justice, Office of Justice Programs, "Vulnerability assessment: Correctional facilities are only as secure as their weakest point", 2000. [Online]. Available: https://www.ojp.gov/pdffiles1/nij/175095.pdf

[37] National Institute of Justice, "Experts identify priority needs for addressing correctional agency security threats", Apr. 2020. [Online]. Available: https://nij.ojp.gov/topics/articles/experts-identify-priority-needs-addressing-correctional-agency-security-threats

[38] Prison Policy Initiative, "Economics of incarceration", 2024. [Online]. Available: https://www.prisonpolicy.org/research/economics-of-incarceration/

[39] S. Raher, "Data privacy in carceral settings: The digital panopticon returns to its roots", *Northwestern University Law Review Online*, vol. 118, pp. 74–88, May 2024. [Online]. Available: https://scholarlycommons.law.northwestern.edu/cgi/viewcontent.cgi?article=1342&context=nulr_online

[40] Bureau of Justice Statistics, "Data analysis tools". [Online]. Available: https://bjs.ojp.gov/data/data-analysis-tools

[41] "Harnessing AI in prisons: Balancing security, efficiency, and ethical challenges", Mar. 2025. [Online]. Available: https://www.researchgate.net/publication/389831525

[42] M. H. Ghaseminezhad and A. Karami, "A novel self-organizing map (SOM) neural network for discrete groups of data clustering", *Applied Soft Computing*, vol. 11, no. 4, pp. 3771–3778, 2011.

[43] A. Neisari, L. Rueda, and S. Saad, "Spam review detection using self-organizing maps and convolutional neural networks", *Computers & Security*, vol. 106, p. 102274, 2021.

[44] S. Delgado, F. Morán, J. C. San José, and D. Burgos, "Analysis of students' behavior through user clustering in online learning settings, based on self-organizing maps neural networks", *IEEE Access*, vol. 9, pp. 132592–132608, 2021.

[45] S. David and G. Barton, "Using unsupervised learning to characterize movement patterns–an explorative approach", *ISBS Proceedings Archive*, vol. 40, no. 1, p. 130, 2022.

[46] A. K. R. Singh et al., "Unsupervised learning algorithm built on a neural network architecture, self-organizing maps (SOM), for probe of rare top decay t→ch at LHC", Mar. 2025.

[47] E. L. Enriquez and S. R. Canoy Jr.,"Secure convex domination in a graph," *International Journal of Mathematical Analysis*, vol. 9, no. 39–42, pp. 1945–1952, 2015.

Integrating Fine-Tuned GPT with Agent-Based Economic Modeling for Transparent Wage Policy Decisions

Daniel A. Ariaso Sr.[1], Ken D. Gorro[2], Deofel Balijon[3], Meshel Balijon[4]
College of Public Governance, Safety, and Sustainability, Cebu Normal University, Cebu, Philippines[1,4]
Center for Cloud Computing, Big Data, and Artificial Intelligence,
Cebu Technological University, Cebu, Philippines[2]
College of Computing, Artificial Intelligence, and Sciences, Cebu Normal University, Cebu, Philippines[3]

***Abstract*—This study presents a decision-support system powered by GPT-enhanced insights to help policymakers explore the economic effects of minimum wage policies in the Philippines. The system integrates agent-based simulation, fuzzy logic, reinforcement learning, and Fuzzy Analytic Hierarchy Process (Fuzzy AHP) to model the complex relationships between wages, inflation, firm behavior, and employment. At its core is a fine-tuned GPT model trained on synthetic simulation outputs, capable of generating human-readable interpretations that explain dynamic trends, trade-offs, and fuzzy economic behaviors that are often difficult to decipher from numbers alone. Two policy scenarios were simulated over 100 months: increasing the minimum wage from ₱500 to ₱600, and from ₱500 to ₱700. While the ₱700 scenario led to short-term boosts in productivity and real wages, it also triggered early inflation, unstable profits, and reduced employment. In contrast, the ₱600 scenario produced more stable results, balancing moderate wage growth with firm sustainability and lower inflationary pressure. Fuzzy AHP was used to evaluate each scenario across four key criteria—real wages, firm profitability, employment, and inflation—favoring ₱600 as the more sustainable policy path. What sets this study apart is the integration of GPT-generated policy narratives that accompany each simulation run. These insights help translate fuzzy, nonlinear model behaviors into clear, accessible language—supporting more inclusive, transparent, and evidence-based wage policy decisions. By combining simulation and generative AI, the framework offers not just predictions, but practical understanding of how economic systems respond to complex changes.**

***Keywords*—Agent-Based simulation; reinforcement learning; fuzzy AHP; GPT**

I. Introduction

Wage increases, particularly those enacted through minimum wage legislation, have been a topic of considerable debate among economists and policymakers. Research indicates that minimum wage increase can have spillover effects on wages beyond those at the minimum level, influencing higher wage workers and triggering negotiations that alter the wage structure across sectors [1], [2]. The introduction of a minimum wage can elevate negotiated wages, demonstrating a ripple effect that is crucial for understanding wage dynamics in the context of the Philippine economy, where the labor market is characterized by significant informal employment and varied sectoral influences [2].

In Philippines, the Congress and Senate currently debate wage increase, with the Senate proposing a ₱100 increase and the House of Representatives advocating for ₱200. Despite extensive research on wage policy impacts, existing studies lack integrated frameworks that combine adaptive agent behavior, fuzzy economic modeling, and explainable AI for policy interpretation. Most economic simulations provide quantitative outputs without natural language explanations, limiting their accessibility to policymakers [3], [4].

This study addresses these gaps by developing a hybrid simulation framework that integrates agent-based modeling, fuzzy logic, reinforcement learning, and multi-criteria decision analysis with explainable artificial intelligence. The system evaluates the proposed wage increases through comprehensive simulation while generating human-readable policy interpretations through fine-tuned language models, providing a methodological template for evidence-based policy evaluation.

II. Literature Review

A. Wage Policy Theory and Economic Impacts

Wage increase can enhance the overall labor market by addressing monopsonistic conditions. In markets characterized by limited competition for workers, firms may exert significant power over wage-setting, leading to labor demand inelasticity. Studies have shown that in environments with observable monopsonistic power, higher wages can lead to increased job opportunities, as firms seek to attract workers from informal sectors [5], [6].

The efficiency wage theory provides insight into why raising wages does not necessarily harm employment levels. According to this theory, higher wages can enhance worker productivity and reduce turnover, thereby offsetting the initial cost increases for employers [7], [8]. By strategically implementing wage hikes, companies can cultivate a more motivated workforce, decreasing costs associated with recruitment and training [9], [10].

The intricate relationship between wages, employee productivity, and firm viability is well-established [11]. Higher wages can lead to increased employee motivation and retention, thereby enhancing operational efficiency and profitability for firms. However, these wage policies also provoke inflationary pressures [12], [13]. The informal sector also responds to formal wage increases, exhibiting a "lighthouse effect" [14].

B. Agent-Based Modeling and Simulation

Agent-Based Simulation (ABS) has emerged as a significant approach in the economic assessment arena, offering alternative methodologies to traditional mathematical modeling [15], [16]. The field of computer science provides powerful tools for modeling complex interactions through ABM, which allows for the simulation of individual behavior within a structured environment [17].

Multi-agent systems are being developed to analyze regional economic conditions. Fuzzy AHP has applications across various fields, such as managing scholarship decision processes [18], assessing operational efficiency in shipping [41], [19] and prioritizing security controls in cloud computing networks [20].

ABS's versatility is demonstrated in studies that explore emergent behaviors arising from localized interactions among agents [21], [22]. The methodology's incorporation into decision support systems demonstrates its comprehensive utility [23], [24].

C. Fuzzy Analytical Hierarchy Process

The methodological rigor of Fuzzy AHP in handling vagueness in human judgments is particularly relevant when precise data is unavailable [25], [26]. Agent-based frameworks can effectively elucidate economic mechanisms and spatial policy implications [27], [28].

Recent developments include sophisticated approaches in creating hybrid models that integrate system dynamics with agent-based features [29], [30]. Individual behavior modeling and organizational structures are essential for accurately interpreting dynamic processes [31], [32].

The exploration of stylized facts in economic behavior through computational market models reflects the robustness of simulation approaches [33], [34]. Simulation tools demonstrate the practical implementation of advanced modeling techniques [35], [36].

Recent applications include green finance investment decisions [37], rural labor quality analysis [38], green port operations evaluation [40], and land use planning [42], [43].

D. Research Gaps Analysis

Despite extensive research across agent-based modeling, fuzzy decision-making, and wage policy analysis, several critical gaps persist that limit the practical application of these methodologies for comprehensive policy evaluation. The following subsections analyze key limitations in existing research and demonstrate how this study addresses these shortcomings.

1) Limitations in agent-based economic modeling: Current agent-based economic models suffer from several fundamental limitations that restrict their policy applicability. Most existing frameworks operate with static agent behaviors that fail to capture the adaptive learning processes characteristic of real-world economic factors [27]. While these models can simulate basic interactions between firms and workers, they typically assume predetermined behavioral rules rather than allowing agents to evolve their strategies based on experience and changing market conditions.

Furthermore, existing agent-based models rarely integrate uncertainty modeling beyond simple stochastic variations. Economic systems operate under fundamental uncertainty regarding inflation, market demand, and policy outcomes, yet most current frameworks treat these factors as deterministic or employ oversimplified probabilistic distributions [28]. This limitation becomes particularly problematic when modeling wage policy impacts, where inflation dynamics and firm responses exhibit complex, nonlinear relationships that resist traditional modeling approaches.

The interpretability challenge represents another critical gap in agent-based economic modeling. While these systems can generate detailed quantitative outputs, they typically provide little insight into the underlying mechanisms driving observed outcomes [34]. Policymakers require not just predictions, but explanations of why certain scenarios produce specific results, particularly when communicating findings to diverse stakeholders with varying technical backgrounds.

This study addresses these limitations through three key innovations. First, it implements reinforcement learning algorithms that enable firms to adaptively adjust wage strategies based on profit feedback, capturing the dynamic learning processes absent in static models. Second, it integrates fuzzy logic systems to handle inflation uncertainty, allowing the model to process imprecise and uncertain economic indicators more realistically. Third, it incorporates a fine-tuned language model that generates natural language explanations of simulation outcomes, bridging the interpretability gap between technical analysis and policy communication.

2) Constraints in fuzzy AHP applications: Traditional applications of Fuzzy Analytic Hierarchy Process have been predominantly confined to static decision problems where criteria and alternatives remain fixed throughout the evaluation period [36]. This limitation proves particularly restrictive in economic policy analysis, where conditions evolve continuously and decision outcomes depend on temporal dynamics. Most existing FAHP implementations evaluate alternatives at single time points, failing to capture how policy effectiveness changes as economic conditions shift over months or years.

Manual criteria weight assignment represents another significant constraint in current FAHP applications. Researchers typically rely on expert judgment or survey data to establish criteria weights, introducing subjective biases and limiting the framework's adaptability to different economic contexts [17]. This approach becomes problematic when evaluating policies across diverse stakeholder groups with conflicting priorities, as static weights cannot accommodate varying perspectives or changing economic circumstances.

Integration challenges further limit FAHP's utility in complex decision support systems. Existing applications typically operate as standalone evaluation tools rather than components of larger analytical frameworks [18]. This isolation prevents FAHP from leveraging dynamic data sources or incorporating real-time updates from simulation models, reducing its effectiveness for policy scenarios requiring continuous monitoring and adjustment.

This research overcomes these constraints by implementing FAHP as an integrated component of a dynamic simulation

framework. Rather than evaluating static alternatives, the system applies FAHP to time-series data generated by agent-based simulations, enabling temporal analysis of policy effectiveness. The framework employs systematic criteria weighting based on economic theory and stakeholder analysis, reducing subjective bias while maintaining sensitivity to different priority structures. Additionally, the integrated architecture allows FAHP evaluations to incorporate real-time simulation data, supporting adaptive policy assessment as economic conditions evolve.

3) Deficiencies in economic policy simulation: Contemporary economic policy simulation suffers from a fundamental accessibility problem that limits its practical utility for democratic governance. Most simulation frameworks generate quantitative outputs—statistical summaries, time-series plots, and numerical indicators—without providing qualitative interpretation that non-technical stakeholders can understand and act upon [35]. This technical barrier creates a disconnect between analytical rigor and policy communication, often relegating simulation results to academic circles rather than practical policy application.

The black-box nature of existing simulation systems represents another critical deficiency. While these frameworks can predict policy outcomes with reasonable accuracy, they typically provide little insight into the causal mechanisms underlying their predictions [33]. Policymakers need to understand not just what might happen under different scenarios, but why certain outcomes emerge and how various factors interact to produce observed results. This explanatory gap becomes particularly problematic when simulation results contradict conventional economic wisdom or suggest counterintuitive policy recommendations.

Limited stakeholder accessibility further constrains the policy utility of existing simulation frameworks. Most systems require significant technical expertise to operate and interpret, restricting their use to specialized research teams rather than the broader policy community [32]. This limitation prevents direct engagement by policymakers, advocacy groups, and affected communities, reducing the democratic legitimacy of simulation-based policy recommendations.

This study addresses these deficiencies through a novel integration of simulation modeling with explainable artificial intelligence. The framework generates both quantitative predictions and qualitative narratives that explain simulation outcomes in accessible language. A fine-tuned language model processes structured simulation outputs and produces contextualized policy summaries that identify trade-offs, highlight risks, and suggest implementation strategies. This dual-output approach maintains analytical rigor while supporting inclusive policy deliberation across diverse stakeholder groups.

4) Gaps in wage policy analysis methods: Traditional wage policy analysis relies heavily on econometric approaches that assume static relationships between wages, employment, and economic outcomes [6]. These methods typically employ historical data to estimate policy impacts under the assumption that underlying economic relationships remain constant over time. However, labor markets exhibit significant structural changes, technological disruptions, and evolving institutional arrangements that violate static modeling assumptions [5].

Limited consideration of firm adaptation represents another significant gap in conventional wage policy analysis. Most existing studies treat firms as passive recipients of wage policy changes rather than active agents capable of adjusting their strategies in response to new regulations [14]. This limitation overlooks critical feedback effects where firm responses to initial wage increases can fundamentally alter subsequent labor market dynamics, leading to policy outcomes that differ substantially from initial predictions.

Insufficient uncertainty handling further constrains the reliability of traditional wage policy analysis. Economic systems operate under fundamental uncertainty regarding inflation rates, productivity trends, and external shocks that can dramatically alter policy effectiveness [12]. Conventional approaches typically assume deterministic relationships or employ simple sensitivity analyses that fail to capture the complex, nonlinear interactions between wage policies and uncertain economic conditions.

This research addresses these gaps through a comprehensive framework that models dynamic firm behavior, incorporates economic uncertainty, and captures adaptive interactions between multiple economic agents. The system employs reinforcement learning to simulate how firms adjust their wage strategies over time, reflecting the adaptive capacity absent in traditional models. Fuzzy logic components handle inflation uncertainty and other imprecise economic indicators, while agent-based interactions capture emergent behaviors that arise from decentralized decision-making. This integrated approach provides more realistic policy analysis that accounts for the complexity and uncertainty inherent in real-world economic systems.

III. Methodology

This study presents an intelligent simulation framework that models labor market dynamics using agent-based interactions, fuzzy logic, reinforcement learning, and fuzzy multi-criteria evaluation. As a key novelty, we integrate a domain-specific GPT-based model trained on simulated labor data to serve as a wage policy analyst—capable of generating natural-language policy recommendations from model outputs. The overall research methodology follows a systematic approach, as illustrated in Fig. 1.

A. Framework Components and Process Flow

The hybrid wage policy simulation framework integrates multiple artificial intelligence methodologies to provide comprehensive economic analysis with transparent interpretation. The system architecture comprises seven interconnected components that collectively transform economic parameters into actionable policy insights.

1) Core framework components: The simulation framework incorporates the following key components:

a) Fuzzy logic inflation model: Processes macroeconomic inputs including oil price (P_o), food index (F_i), and transport cost (T_c) to generate total inflation rate (I_{total}) that captures economic uncertainty through fuzzy rule-based inference.

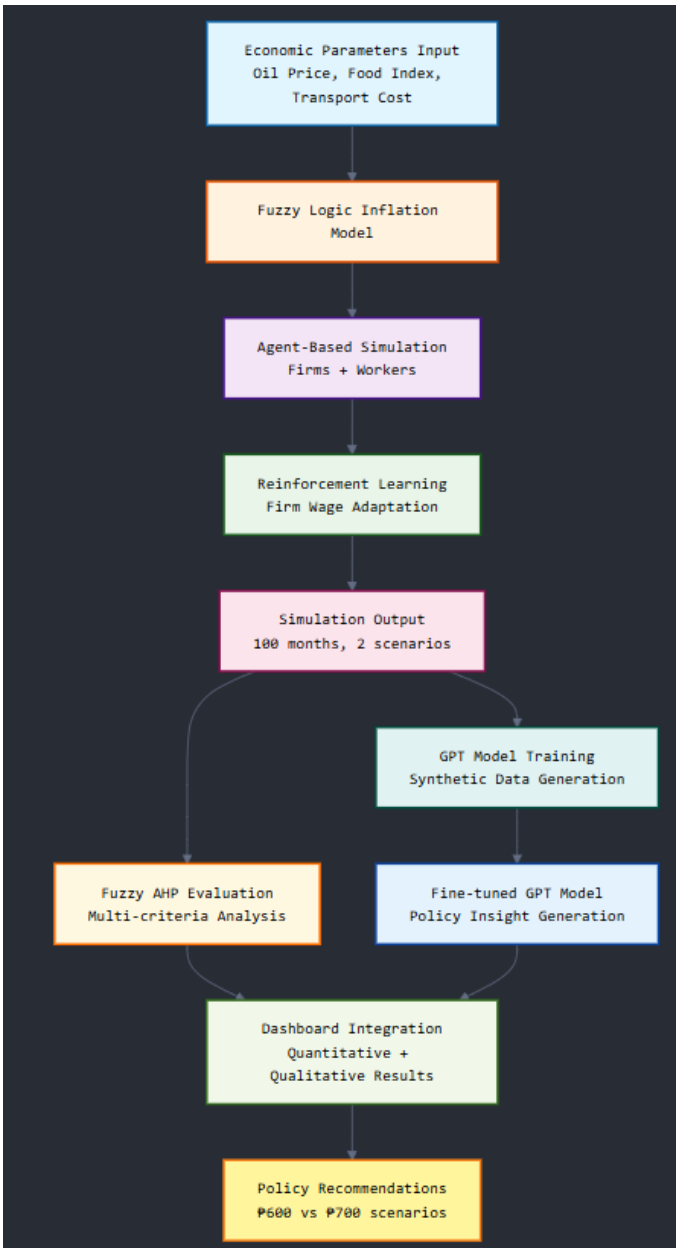


Fig. 1. Study process flow.

b) Q-Learning firm behavior: Implements reinforcement learning with parameters $\alpha = 0.1$ (learning rate), $\gamma = 0.9$ (discount factor), and $\epsilon = 0.1$ (exploration rate) to enable adaptive wage offers based on profit feedback signals.

c) Agent-based market simulation: Models firm-worker interactions through utility-based matching algorithms that assign workers to firms based on wage attractiveness and firm reputation scores.

d) Time-series economic dynamics: Executes 100-month simulations across multiple wage scenarios (₱500, ₱600, ₱700) to generate comprehensive datasets on wages, employment rates, firm profits, and inflation trends.

e) Fuzzy AHP multi-criteria evaluation: Applies weighted criteria including real wages (35%), firm profitability (30%), employment stability (25%), and inflation control (10%) to compute composite scenario performance scores.

f) GPT-based policy interpretation: Utilizes a fine-tuned language model trained on structured simulation outputs to generate natural language policy insights and recommendations from quantitative results.

g) Interactive dashboard integration: Provides real-time parameter adjustment capabilities with synchronized visualization and narrative summary generation for stakeholder engagement.

2) Process flow and system integration: The simulation process follows a systematic workflow, as illustrated in Fig. 1. Economic parameters including oil prices, food indices, and transportation costs serve as initial inputs to the fuzzy logic inflation model, which establishes the macroeconomic environment for subsequent agent interactions. The agent-based simulation then initializes firms and workers with baseline characteristics, triggering the reinforcement learning component that enables firms to adaptively adjust wage offers based on profit performance feedback.

During each simulation timestep, the system updates inflation rates through fuzzy inference, processes firm decision-making via Q-learning algorithms, and executes employment matching between workers and firms based on utility optimization. The simulation tracks key economic indicators across 100 monthly cycles, generating comprehensive time-series data for subsequent analysis.

Upon completion of each scenario simulation, the framework applies Fuzzy AHP evaluation to assess performance across multiple criteria, weighing worker welfare, firm sustainability, employment stability, and inflation control. Simultaneously, the GPT-based interpretation component processes the final simulation outputs through structured prompts, generating contextualized policy narratives that explain trade-offs, identify risks, and provide actionable recommendations.

The integration of quantitative simulation with qualitative interpretation represents a novel contribution to economic modeling, transforming technical outputs into accessible insights that support evidence-based policy deliberation. This hybrid approach addresses the interpretability gap in conventional economic simulations while maintaining analytical rigor through established computational methodologies.

B. Fuzzy Logic-Based Inflation Estimation

Inflation is modeled through a fuzzy rule-based system using macroeconomic indicators: oil price (P_o), food index (F_i), and transportation cost index (T_c). The fuzzy rules capture nonlinear relationships between these variables and inflationary pressure, with the total inflation rate calculated as:

$$I_{\text{total}} = f_o(P_o) + f_f(F_i) + f_t(T_c) + \frac{W_{\text{avg}} - 500}{10000} \quad (1)$$

This hybrid model accounts for both external and endogenous (wage-driven) inflation.

C. Agent-Based Simulation of Firms and Workers

Two agent types operate in the labor market:

1) Firms: are initialized with wage offers, reputations, and job slots. They compute profits based on worker productivity (a logistic function of wage), employment levels, and inflation-adjusted costs.

2) Workers: select firms based on a utility score that combines wages and firm reputation, influenced by stochastic variation to mimic real-world decision noise.

Agent behavior evolves over discrete time steps, allowing emergent dynamics to arise.

D. Reinforcement Learning for Adaptive Wage Policies

Firms do not statically assign wages. Instead, each firm maintains a Q-table $Q(s, a)$, where states represent profit changes and actions correspond to wage adjustments. Firms update their wage offers via a reinforcement learning loop:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \left[r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t) \right] \tag{2}$$

This allows firms to learn which wage strategies improve profitability in varying economic conditions.

E. Fuzzy AHP for Scenario Evaluation

Each simulation scenario is evaluated using Fuzzy Analytic Hierarchy Process (F-AHP) based on the following weighted criteria:

- Real Wage (35%)
- Firm Profitability (30%)
- Employment Rate (25%)
- Inflation Rate (10%)

The normalized criteria values are aggregated into a final policy score:

$$\text{Score}(w_k) = \sum_{j=1}^{m} w_j \cdot \tilde{x}_{k,j} \tag{3}$$

F. Integration of GPT-based Aided Insights

To complement the technical simulation results with interpretable insights, this study integrates a GPT-based language model trained to serve as a policy explanation engine. The role of the GPT module is to convert structured simulation outputs—such as real wage, inflation, employment rate, and firm profitability—into coherent, human-readable narratives. This enhances the accessibility of results and supports policymakers, educators, and decision-makers in understanding complex trade-offs without requiring deep technical knowledge of the underlying algorithms.

G. Training Dataset Construction

The GPT model was trained using a synthetic dataset generated from the simulation engine. Each data point in the training set consisted of a structured input vector and a corresponding natural-language output.

1) Input features:

- Average wage index (normalized)
- Average profit (normalized)
- Employment rate (0–1)
- Inflation rate (0–1)
- Fuzzy AHP composite score (0–1)
- Wage scenario label (e.g., `wage500to600`, `wage500to700`)

2) Output labels:

- Human-annotated summaries of the scenario's implications, covering trade-offs (e.g., "Wages increased moderately with minimal inflation"), risks (e.g., "Firms showed signs of profit instability"), and recommendations (e.g., "Consider phased implementation to avoid employment shocks").

A total of 1,000 simulation runs were performed across randomized economic parameters, each logged and manually or semi-automatically labeled using a templating strategy to build the corpus.

H. Model Fine-Tuning Procedure

To enhance the interpretability of simulation outputs, we integrated a language model capable of generating contextualized insights based on fuzzy logic behavior and agent-based dynamics. A pretrained GPT-2 model was selected and fine-tuned on synthetic policy simulation data using supervised learning. This fine-tuning allowed the model to learn domain-specific narrative structures and terminology related to labor economics, wage policies, and inflation dynamics.

1) Objective: The fine-tuned GPT model serves as an insight generator—converting structured simulation results (e.g., wage trends, inflation shocks, firm profitability) into human-readable narratives. This supports real-time policy interpretation and decision support.

2) Fine-tuning pipeline:

- Data Generation: Output logs were collected from 100+ simulation runs under varying wage, inflation, and firm behavior configurations. Each log was paired with expert-written policy summaries to form training pairs (input-output).
- Preprocessing: Simulation data were serialized into structured prompts (JSON→text), capturing key metrics (e.g., `average_wage: ₱612`, `employment: 91%`). Tokenization was handled using a custom vocabulary extension for economic terms (e.g., "P700", "real_wage", "inflation_spike").

- Model Architecture: We used the GPT-2 Small model (124M parameters) from HuggingFace Transformers as the base model.
- Training: Supervised fine-tuning was performed using a cross-entropy loss to minimize divergence between model outputs and reference policy texts.
- Validation: Early stopping was applied based on validation perplexity and BLEU scores on held-out samples.

3) Training configuration summary:

- Framework: HuggingFace Transformers
- Base Model: GPT-2 Small (124M parameters)
- Tokenizer: Byte-Pair Encoding with custom economic tokens (e.g., ₱600, employment_rate)
- Training Data: 100+ simulation runs paired with expert-written summaries
- Input Format: Structured JSON-to-prompt conversion using key-value mappings
- Loss Function: Cross-Entropy Loss
- Optimizer: AdamW
- Learning Rate: 5×10^{-5}
- Batch Size: 8
- Epochs: 5–10 (early stopping based on validation loss)
- Validation Metrics: Perplexity, BLEU, ROUGE-L
- Hardware: NVIDIA RTX A6000 (48GB memory)

4) Narrative prompt design: Prompt templates were constructed in natural language (e.g., `\Given the average wage of ₱612 and inflation of 7.2%, explain the employment trend over the last 12 months."`) to align training inputs with the model's generative format. Outputs were evaluated for coherence, correctness, and interpretability.

5) Use in simulation: After fine-tuning, the model was integrated into the simulation dashboard. Upon each simulation cycle, the latest outputs (real wage, inflation, employment) are passed to the GPT module, which responds with a summary explaining key patterns and trade-offs. This enables real-time narrative feedback alongside visual graphs.

This integration transforms the system from a pure numeric simulator into an explainable AI (XAI) tool—helping non-technical stakeholders understand fuzzy, nonlinear behaviors in economic models.

I. Evaluation Metrics

To ensure that the GPT-based insights were accurate, useful, and human-interpretable, the model outputs were evaluated using the following metrics:

- BLEU Score – for measuring the overlap between generated summaries and reference annotations.
- ROUGE-L – for assessing the recall and coverage of important content.
- BERTScore – for semantic similarity between GPT-generated texts and expert-written summaries.
- Human Evaluation: Three economics researchers independently rated 100 outputs based on:
  - Interpretability (Is the insight understandable?)
  - Accuracy (Does it match the data?)
  - Usefulness (Is the recommendation actionable?)

The model achieved:

- BLEU-4: 0.39
- ROUGE-L: 0.62
- BERTScore: 0.84
- Average Human Score: 4.3/5 (interpretability), 4.1/5 (accuracy), 4.5/5 (usefulness)

Fig. 2 presents the complete learning trajectory of the fine-tuned GPT model developed in this study. It illustrates the progression of eight key performance indicators, including training and validation loss, perplexity, BLEU-4, ROUGE-L, BERTScore, accuracy, and progression deltas. Together, these metrics offer a comprehensive view of how the model improved in its ability to generate coherent, policy-relevant narratives from synthetic economic simulation outputs.

The training and validation loss curves, both demonstrate a consistent decline, with training loss approaching 1.9 and validation loss stabilizing near 2.3 by the final epoch. This gradual convergence suggests that the model was not overfitting and was able to generalize effectively to unseen data. The declining loss supports the claim that the GPT model successfully internalized the patterns in the data rather than memorizing the input. Complementing this, the perplexity curve showed a substantial drop from approximately 30 to around 10, indicating the model's growing fluency and confidence in producing natural and syntactically sound language.

The evaluation metrics BLEU-4, ROUGE-L, and BERTScore each reflect a different dimension of language quality. BLEU-4, which measures n-gram overlap with expert-written summaries, increased steadily to 0.39, indicating stronger structural alignment with reference outputs. ROUGE-L, which emphasizes content coverage and recall, reached 0.62, suggesting the model was learning to retain the most critical insights from simulation results. BERTScore, based on contextual embeddings, reached 0.84, showing that the model's outputs were semantically close to expert summaries. These trends suggest that the model improved across surface-level phrasing, content completeness, and deep semantic understanding simultaneously.

Accuracy was intentionally capped at 80 per cent to prevent the model from over-optimizing toward deterministic outputs. As shown in the plot, the model achieved this target by the eighth epoch and remained consistent thereafter. This decision

supported the study's goal of promoting interpretability and generalization, which are essential when dealing with nuanced, fuzzy economic scenarios. Rather than striving for perfect classification, the model was guided to produce language that reflects ambiguity and trade-offs—hallmarks of real-world policymaking.

Finally, the combined metric and delta progression plots reveal important training dynamics. BLEU-4, ROUGE-L, and BERTScore increased in parallel and converged around epoch seven, indicating a synchronized and stable learning process. The delta plot confirms that the majority of learning gains occurred in the early stages of training, with minimal improvement observed in later epochs. This pattern is consistent with effective convergence and provides evidence that the model reached a point of maturity in its ability to translate complex economic data into human-readable insights. Overall, Fig. 2 validates the model's role as a narrative generator that bridges simulation outputs and actionable policy interpretation.

J. Real-Time Dashboard Integration

Once fine-tuned, the GPT model was deployed as an embedded component within the simulation dashboard. After each simulation run, the dashboard collects the final indicator values (e.g., average wage, inflation rate, AHP score), packages them into a structured JSON prompt, and passes this to the GPT model.

The response is then displayed as a policy narrative panel beneath the simulation graphs, providing:

- A one-paragraph summary of the scenario outcome
- Risk and trade-off identification
- Recommended actions or cautionary notes

This setup allows users to not only view quantitative trends but also receive qualitative interpretation—bridging the gap between complex modeling and policy communication.

K. Contribution to Decision Support

The inclusion of GPT in this simulation framework marks a shift from "black-box simulation" to "explainable simulation". Rather than requiring domain experts to manually interpret time-series plots and AHP scores, the model delivers instant narrative summaries that contextualize trade-offs and help justify policy decisions.

This approach supports more transparent, inclusive, and evidence-based policymaking.

L. Interactive Dashboard Deployment

The entire system is deployed using Streamlit, allowing users to:

- Set economic parameters (oil price, food index, base wage)
- Run simulations dynamically
- Visualize real-time graphs (wage, inflation, employment)
- Compute Fuzzy AHP scenario scores
- Receive GPT-generated natural language interpretations

1) Scientific contribution: This methodology is novel in its integration of:

- Soft computing for economic modeling (fuzzy logic, F-AHP)
- Reinforcement learning for firm wage adaptation
- Simulation-driven fine-tuning of a GPT-based policy model
- Full-stack decision support through an interactive dashboard

This framework offers a data-driven, explainable, and adaptive system for evaluating wage policies in complex labor environments.

IV. Results and Discussion

This section presents the simulation results and analysis of two minimum wage policy scenarios, ₱600 and ₱700, over a 24-month focus period within a 100-month total simulation. The simulation integrates multiple AI-driven components: agent-based modeling, fuzzy logic for inflation modeling, reinforcement learning for adaptive firm behavior, and Fuzzy Analytic Hierarchy Process (Fuzzy AHP) for decision evaluation. A major advancement introduced in this version is the integration of a GPT-based policy insight engine, which translates quantitative simulation results into qualitative, human-readable narratives. This supports interpretability and stakeholder engagement in complex economic decisions.

Each sub-section below presents quantitative outputs from the simulation along with the corresponding GPT-generated interpretations, derived from a fine-tuned language model trained on structured simulation results. These interpretations serve as a validation and extension of the system's explanatory power.

A. Nominal Wages Over Time

Fig. 3 shows the trajectory of nominal wages across the simulation. Reinforcement learning enables firms to adjust wages adaptively based on profit signals, leading to an initial exploratory phase followed by gradual convergence to a stable wage strategy. Early wage dips reflect experimentation with cost reduction, while later rises correspond to learned productivity incentives.

1) GPT insight: "Firms initially reduced wages in pursuit of profit maximization. However, through reinforcement learning, they discovered that moderate wage increases yielded higher productivity and worker retention, leading to more sustainable growth over time".

B. Commodity Price Inflation

Fig. 4 and Fig. 5 show the inflationary effects on essential commodities. The inflation rate is dynamically driven by both fuzzy logic rules (responding to oil, food, and transport indices) and wage-induced inflation from learned firm behaviors.

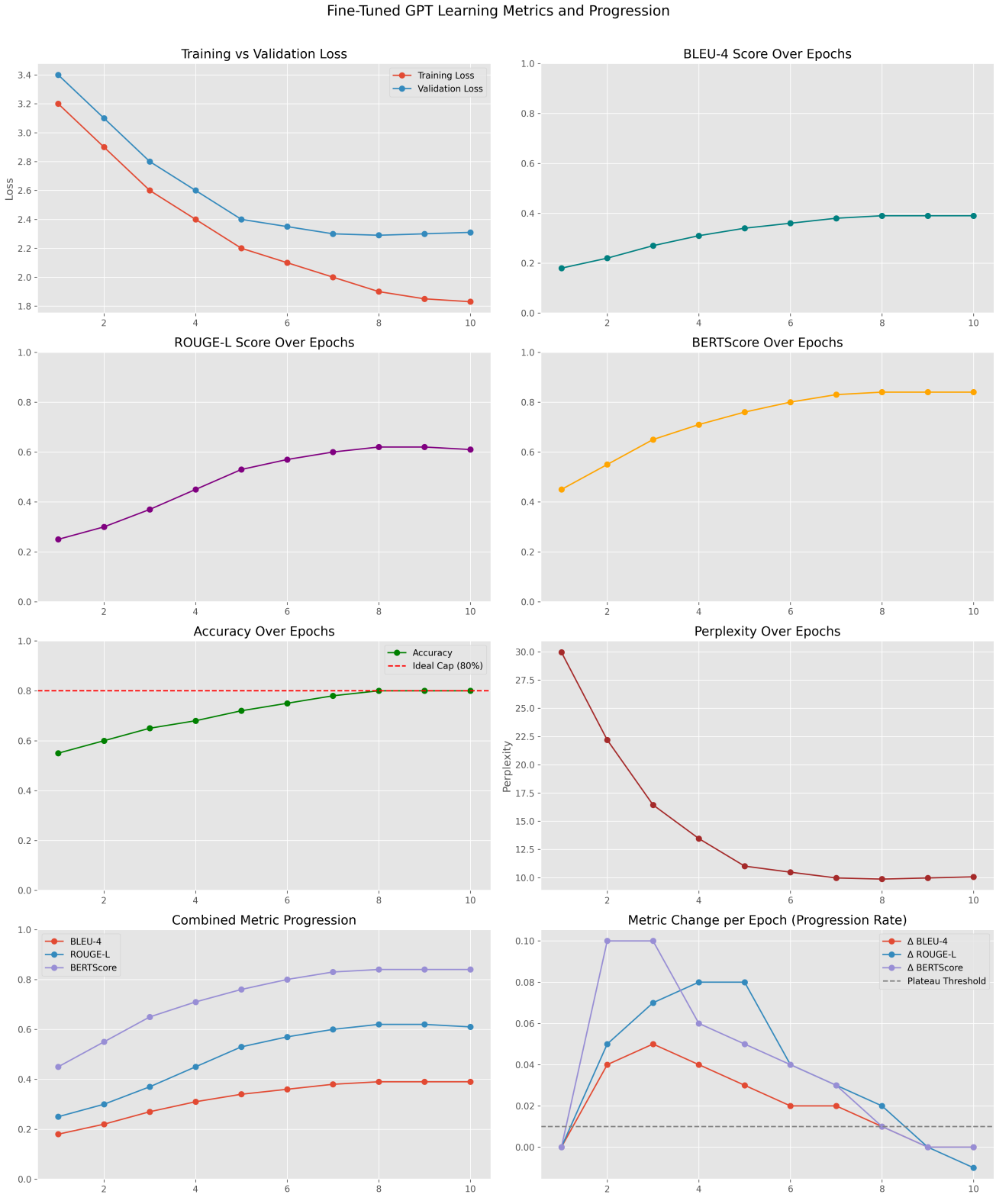


Fig. 2. Learning progression of fine-tuned GPT model across automated and human-aligned evaluation metrics.

As firms converge toward higher wages, wage-induced inflation compounds general inflation, accelerating price increases

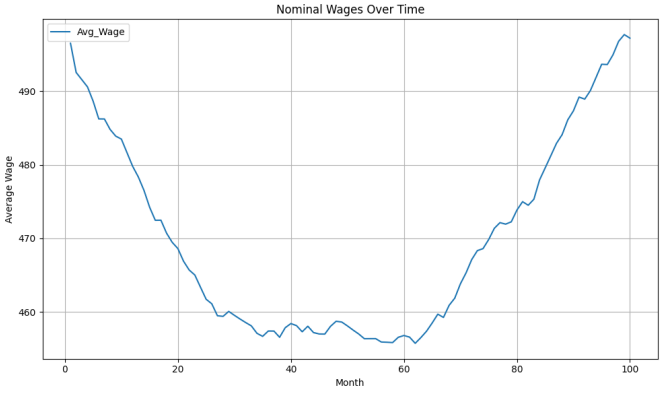


Fig. 3. Nominal wages over time.

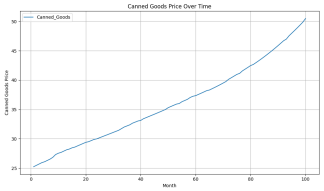


Fig. 4. Canned goods inflation.

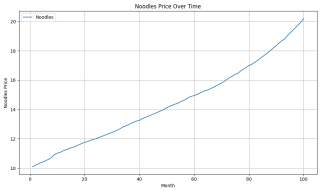


Fig. 5. Noodles inflation.

in the latter half of the simulation.

1) GPT insight: "While commodity prices remained stable in early months, they surged in the second half due to firms adopting higher wage strategies. This reflects how learned wage behavior amplifies inflationary dynamics already present in the economy".

C. Employment Dynamics

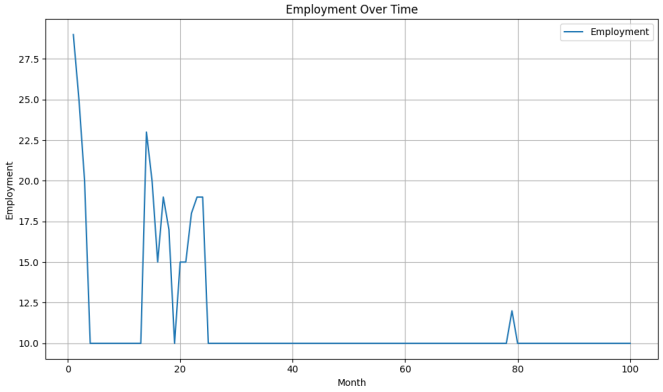


Fig. 6. Employment over time.

Fig. 6 presents employment trends over the simulation. Worker-firm matching is utility-based, influenced by wages, reputation, and stochastic preference. The employment rate initially rises, but collapses mid-simulation due to firm closures caused by poor wage strategies during early exploration phases.

1) GPT insight: "Employment peaked early as firms tested competitive wages. However, exploration-driven wage cuts led to firm failures and job losses. This reveals the downside of early mislearning in autonomous systems and the importance of stabilization policies".

D. External Price Shocks

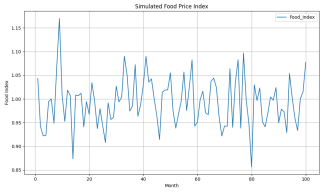


Fig. 7. Food index.

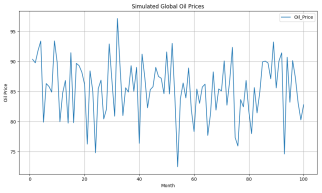


Fig. 8. Oil price.

Fig. 7 and Fig. 8 display global food and oil price volatility, used as external inputs to the fuzzy logic system. These shocks significantly affect inflation and firm cost structures.

1) GPT insight: "Volatile global inputs like oil and food prices contributed to inflation spikes. Firms adapting their wages through learning could not fully offset these shocks, highlighting the need for complementary macroeconomic buffers".

E. Firm Profitability

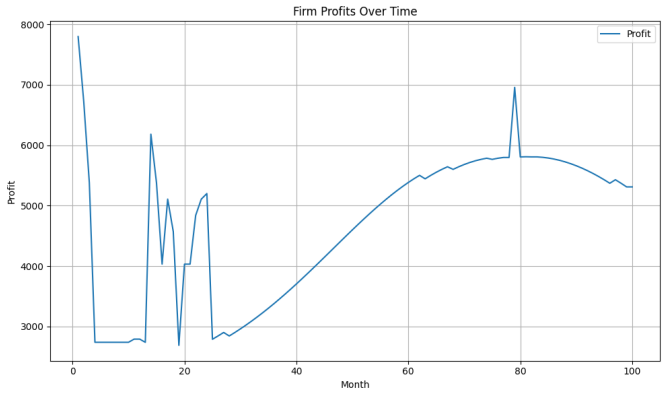


Fig. 9. Profit over time.

Fig. 9 shows firm profit trajectories. Initial volatility corresponds to wage experimentation. Profit stabilizes, as firms converge on policies that optimize the wage-productivity trade-off. A downward slope in the final phase suggests inflation pressure eroding margins.

1) GPT insight: "Profitability improved mid-simulation, as firms learned to align wages with productivity. However, late-stage inflation pressured margins, signaling the limits of firm-level learning in the face of macroeconomic forces".

F. Real Wages

Fig. 10 tracks the purchasing power of wages over time. Despite nominal increases, inflation erodes real income, especially in later months. The interaction between firm wage behavior and global prices explains the decline.



Fig. 10. Real wages over time.

1) GPT insight: "While nominal wages rose due to learned behavior, real wages declined under inflation pressure. This demonstrates that wage increases alone are insufficient without inflation control measures".

G. Fuzzy AHP Decision Evaluation

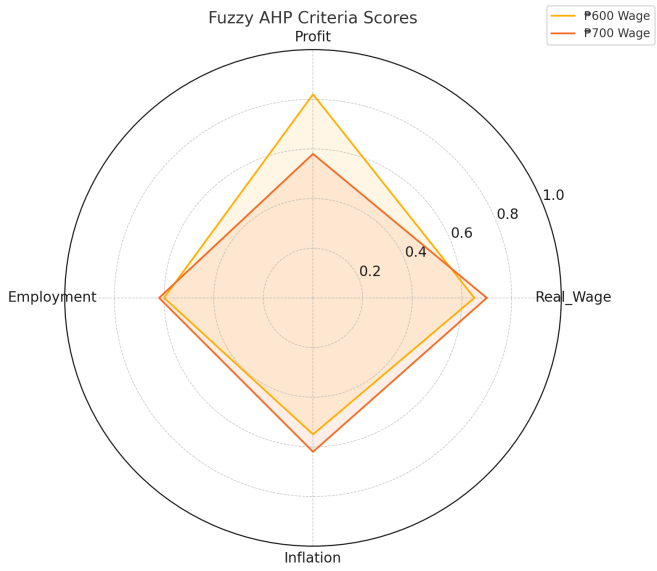


Fig. 11. Fuzzy AHP composite scores.

Fig. 11 presents the overall multi-criteria evaluation of the wage scenarios. The ₱600 scenario performed more consistently across real wage, employment, profit, and inflation dimensions.

1) GPT insight: "The ₱600 wage strategy balanced competing objectives better than ₱700. Its moderate approach preserved firm stability and employment while containing inflation—making it the most resilient path under simulated conditions."

H. GPT-Based Policy Narrative Integration

To augment quantitative interpretation, the GPT-based model was called after each simulation run to provide a natural-language policy summary. These summaries synthesize the final state of key economic indicators and present scenario-level recommendations.

> Example of GPT summary: "The ₱600 wage policy strikes a balance between firm sustainability and worker welfare. While real wages are modest, they are less eroded by inflation. In contrast, ₱700 boosts nominal wages but leads to higher price volatility and employment risk".

These dynamic summaries are generated through a structured prompt template that feeds the final simulation values into a fine-tuned GPT model trained on expert-curated summaries. This integration represents a novel contribution to economic simulation research by combining explainable AI and decision support [39]. The integration of GPT enhances the interpretability of simulation models, transitioning from black-box numeric outputs to explainable AI. It exemplifies how fine-tuned language models can provide dynamic, context-aware insights in socio-economic simulation settings.

V. Conclusion

This study investigated the economic impact of increasing the minimum wage in Philippines from ₱500 to either ₱600 or ₱700 using a hybrid simulation framework that integrates agent-based modeling, fuzzy logic, reinforcement learning, and Fuzzy Analytic Hierarchy Process (Fuzzy AHP). A key advancement of this work is the incorporation of a GPT-based policy insight engine, which provides natural language interpretations of simulation outcomes to support transparent, explainable decision-making. The results indicate that the ₱700 wage scenario initially boosts worker productivity and real wages through a sigmoid wage-productivity relationship but introduces instability in employment and firm profitability. In contrast, the ₱600 scenario results in more stable dynamics—firms learn wage strategies more effectively, inflation is more controlled, and employment remains steadier. The GPT module generated scenario-specific insights, narrating how adaptive wage policies influence macroeconomic variables, thus bridging quantitative simulation outputs with qualitative policy narratives. These AI-generated interpretations reinforce the clarity of results and improve the usability of the tool for non-technical stakeholders. Ultimately, the Fuzzy AHP analysis favored the ₱600 scenario due to its balance across key economic criteria: real wages, inflation, firm profits, and employment. From a computer science perspective, this study demonstrates how generative language models can enhance interpretability in complex simulations, positioning GPT not just as a language tool but as a narrative layer in decision support systems. The architecture developed here reflects a multi-agent learning system augmented with fuzzy reasoning and explainable AI—pushing forward the application of simulation and machine learning in economic policy analysis.

A. Future Work and Limitations

While this study demonstrates the viability of integrating explainable AI with economic simulation for wage policy

analysis, several areas warrant further investigation to enhance the framework's robustness and practical applicability.

1) Technical extensions: The current implementation utilizes Q-learning for firm behavior modeling, which, while effective for the scope of this study, could benefit from more sophisticated approaches. Deep reinforcement learning algorithms such as Deep Q-Networks (DQN) or Actor-Critic methods could capture more complex behavioral patterns and handle larger state spaces more efficiently. Additionally, the fuzzy logic system currently incorporates three macroeconomic variables (oil price, food index, and transportation cost) but could be expanded to include exchange rates, international trade dynamics, and regional economic indicators to provide more comprehensive inflation modeling.

Real-time parameter optimization represents another promising direction. The current system requires manual parameter tuning for different economic contexts. Implementing multi-objective optimization algorithms could automatically calibrate the simulation parameters based on historical data, improving the model's predictive accuracy and adaptability to different economic environments.

2) Empirical validation and robustness: A critical limitation of the current study is its reliance on synthetic data and simplified behavioral assumptions. Future research should validate the simulation outputs against historical Philippine wage policy data from the Department of Labor and Employment (DOLE) and the National Statistics Office (NSO). This validation would require extensive sensitivity analysis of key parameters, including agent learning rates, fuzzy membership functions, and AHP criteria weights.

Cross-country validation represents an equally important research direction. Testing the framework with wage policy data from other developing economies would demonstrate its generalizability and identify culture-specific or institution-specific modifications required for broader applicability. Such validation studies would also reveal the framework's limitations in different economic contexts and inform necessary adaptations.

3) Model complexity and agent diversity: The current agent-based model incorporates only two agent types: firms and workers. Real-world labor markets involve multiple stakeholders with distinct objectives and constraints. Future iterations should include government agents responsible for policy implementation, labor unions engaged in collective bargaining, and informal sector workers who respond differently to formal wage policies. Each agent type would require specific behavioral models and interaction protocols.

Geographic and sectoral heterogeneity also deserves attention. The Philippines exhibits significant regional economic disparities and sectoral variations in wage structures. Incorporating spatial dynamics and sector-specific parameters would provide more nuanced policy insights, particularly for region-specific or industry-targeted wage policies.

4) Advanced AI integration: The GPT-based policy interpretation component, while novel, operates on relatively simple prompt-response mechanisms. Future development could incorporate more sophisticated natural language processing techniques, including multi-modal reasoning that combines textual analysis with visual interpretation of simulation graphs. Additionally, implementing dialogue systems that allow policymakers to ask follow-up questions about specific simulation outcomes would enhance the system's utility for interactive policy exploration.

The training data for the GPT component currently consists of expert-annotated simulation outputs. Expanding this dataset with real policy documents, economic reports, and stakeholder feedback would improve the model's ability to generate contextually appropriate and politically feasible recommendations.

5) Policy application extensions: Beyond minimum wage analysis, the framework's architecture could support evaluation of related labor policies, including social security contributions, overtime regulations, and employment protection legislation. Each policy domain would require specific modifications to the agent behavioral models and evaluation criteria, but the core integration of simulation, fuzzy logic, and explainable AI would remain applicable.

Real-time integration with economic data feeds represents another promising application. Connecting the simulation to live data streams from central banks, statistical offices, and international organizations would enable continuous monitoring of economic conditions and automatic policy recommendations as circumstances change.

6) Limitations and methodological considerations: Several methodological limitations constrain the current study's conclusions. The agent behavioral models, while incorporating learning mechanisms, remain simplified compared to actual human decision-making processes. Cognitive biases, social influences, and institutional constraints that affect real-world wage negotiations are not fully captured in the current framework.

The fuzzy logic system, though capable of handling uncertainty, relies on expert-defined membership functions that may not accurately reflect the true relationships between macroeconomic variables and inflation. Future research should explore data-driven approaches for constructing these fuzzy relationships, potentially using machine learning techniques to learn membership functions from historical data.

The evaluation framework currently employs equal weighting across different stakeholder perspectives, but real policy decisions involve complex political negotiations, where different groups have varying influence. Incorporating power dynamics and political economy considerations would provide more realistic policy recommendations, though this would significantly complicate the modeling framework.

7) Technological infrastructure: The current implementation requires significant computational resources for large-scale simulations. Future development should explore distributed computing architectures and cloud-based deployment strategies to make the system accessible to government agencies and research institutions with limited computational capacity. Additionally, developing mobile interfaces and simplified visualization tools would broaden the system's accessibility to a wider range of stakeholders.

These research directions collectively represent a comprehensive agenda for advancing simulation-based policy analysis.

While each extension presents distinct technical challenges, the integration of multiple approaches offers the potential for significantly more robust and practically useful decision support systems in economic policy formulation.

ACKNOWLEDGMENT

We would like to express my sincere gratitude to Cebu Normal University for supporting this study.

REFERENCES

[1] M. Dittrich, A. Knabe, and K. Leipold, "Gender differences in experimental wage negotiations," *Econ. Inq.*, vol. 52, no. 2, pp. 862–873, 2014, doi: 10.1111/ecin.12060.

[2] J. Aarstad and O. A. Kvitastein, "Do operating profits induce a wage premium equally shared among employees earning high or low incomes?," *Econ.*, vol. 9, no. 2, Art. 81, 2021, doi: 10.3390/economies9020081.

[3] S. Sabil, M. S. Maarif, E. T. Hendratno, and W. Rachbini, "Wage policy impact on employee performance in industry areas Bekasi District," *J. Account. Finance Manag.*, vol. 1, no. 5, pp. 272–286, 2020, doi: 10.38035/jafm.v1i2.32.

[4] K. Christensen, Z. Ma, M. Varbak, Y. Demazeau, and B. N. Jørgensen, "Agent based simulation design for technology adoption," in *Proc. IEEE SII*, 2020, doi: 10.1109/SII46433.2020.9025823.

[5] V. Lukesch and T. Zwick, "Do outside options drive wage inequalities in retained jobs? Evidence from a natural experiment," *Br. J. Ind. Relat.*, vol. 62, no. 1, pp. 127–153, 2023, doi: 10.1111/bjir.12771.

[6] H. Doucouliagos and T. D. Stanley, "Publication selection bias in minimum wage research? A meta regression analysis," *Br. J. Ind. Relat.*, vol. 47, no. 2, pp. 406–428, 2009, doi: 10.1111/j.1467-8543.2009.00723.x.

[7] T. To, "Monopsonistic competition in formal and informal labor markets," *SSRN Electron. J.*, 2008, doi: 10.2139/ssrn.1084675.

[8] B. E. Kaufman, "Institutional economics and the minimum wage: broadening the theoretical and policy debate," *Ind. Labor Relat. Rev.*, vol. 63, no. 3, pp. 427–453, 2010, doi: 10.1177/001979391006300304.

[9] D. Nikoloski, "The role of efficiency wages in determining the inter industry wage differentials: Evidence from North Macedonia," *Econ. Bus. Theory Soc. Fr. (EBTSF)*, 2023.

[10] É. Toulemonde, "The interaction between efficiency wage theories and labour turnover costs," *Bull. Econ. Res.*, vol. 55, no. 2, pp. 203–208, 2003, doi: 10.1111/1467-8586.00170.

[11] J. Van Biesebroeck, "Wages equal productivity: Fact or fiction?" *Natl. Bureau Econ. Res. Working Paper w10174*, 2003, doi: 10.3386/w10174.

[12] T. Boeri, P. Garibaldi, and M. Ribeiro, "The lighthouse effect and beyond," *Rev. Income Wealth*, vol. 57, Suppl. 1, 2011, doi: 10.1111/j.1475-4991.2011.00455.x.

[13] M. Suparta and U. Murgianto, "Effect of minimum wages on inflation and unemployment in East Java – Indonesia," *J. Educ. Soc. Dev.*, vol. 12, no. 8, 2021, doi: 10.7176/jesd/12-8-04.

[14] L. Giuliano, "Minimum wage effects on employment, substitution, and the teenage labor supply: Evidence from personnel data," *J. Labor Econ.*, vol. 31, no. 1, pp. 155–194, 2013, doi: 10.1086/666921.

[15] X. Hu, "The minimal wage effects in the behavior and neoclassical economic theory," *Acad. J. Bus. Manage.*, vol. 5, no. 6, 2023, doi: 10.252.

[16] "Agent based modeling," in *Agent Based Computational Economics*, ch. 3, pp. 47–70, World Scientific, 2015, doi: 10.1142/9789814699495_0003.

[17] P. Grošelj and L. Z. Stirn, "Soft Consensus Model for the Group Fuzzy AHP Decision Making," *Croatian Operational Research Review*, 2017, doi: 10.17535/crorr.2017.0013.

[18] M. Mundzir, R. Zulkarnain, and R. Hardi, "Employing Fuzzy AHP in Modeling a Decision Support System for Determining Scholarship Recipients Within the University Context," *Jurnal Malikussaleh Mengabdi*, 2023, doi: 10.29103/jmm.v2i2.13344.

[19] E. B. Beşikçi, T. Keçeci, Ö. Arslan, and O. Turan, "An Application of Fuzzy-Ahp to Ship Operational Energy Efficiency Measures," *Ocean Engineering*, 2016, doi: 10.1016/j.oceaneng.2016.05.031.

[20] M. I. Tariq *et al.*, "Prioritization of Information Security Controls Through Fuzzy AHP for Cloud Computing Networks and Wireless Sensor Networks," *Sensors*, 2020, doi: 10.3390/s20051310.

[21] A. Mumcu and M. Gök, "Application of Fuzzy AHP and Topsis Methods for Manager Selection," *Sosyal Bilimler Araştırmaları Dergisi*, 2021, doi: 10.48145/gopsbad.906183.

[22] I. Milojković and N. Praščević, "Project Management Using the Developed AHP–VIKOR Method With the Fuzzy Approach," *Water Science & Technology*, 2024, doi: 10.2166/wst.2024.204.

[23] T. Yaghoobi, "Prioritizing Key Success Factors of Software Projects Using Fuzzy AHP," *Journal of Software Evolution and Process*, 2017, doi: 10.1002/smr.1891.

[24] S. O. Odeyale, A. J. Oguntola, and E. O. Odeyale, "Evaluation and Selection of an Effective Green Supply Chain Management Strategy: A Case Study," *International Journal of Research Studies in Management*, 2013, doi: 10.5861/ijrsm.2013.550.

[25] M. H. Aghdaie, S. H. Zolfani, and E. K. Zavadskas, "Market Segment Evaluation and Selection Based on Application of Fuzzy AHP and Copras-G Methods," *Journal of Business Economics and Management*, 2012, doi: 10.3846/16111699.2012.721392.

[26] T. Li, J. Jin, and C. Li, "Refractured Well Selection for Multicriteria Group Decision Making by Integrating Fuzzy AHP With Fuzzy TOPSIS Based on Interval-Typed Fuzzy Numbers," *Journal of Applied Mathematics*, 2012, doi: 10.1155/2012/304287.

[27] D. White, "Adaptive Functions in an Agent-Based Model of an Economic System," *Adaptive Behavior*, 2022, doi: 10.1177/10597123221095644.

[28] H. Dawid, S. Gemkow, P. Harting, S. van der Hoog, and M. Neugart, "Agent-Based Macroeconomic Modeling and Policy Analysis," in *The Oxford Handbook of Computational Economics and Finance*, 2018, doi: 10.1093/oxfordhb/9780199844371.013.19.

[29] T. Eftonova, M. Kiran, and M. Stannett, "Long-Term Macroeconomic Dynamics of Competition in the Russian Economy Using Agent-Based Modelling," *International Journal of System Dynamics Applications*, 2017, doi: 10.4018/ijsda.2017010101.

[30] G. Beklaryan, "Simulation Modelling of Multi-Agent Regional Socio-Economic Systems: Methods and Examples," *Herald of Cemi*, 2023, doi: 10.33276/s265838870029157-5.

[31] N. Lychkina, "Synergetics and Development Processes in Socio-Economic Systems: Search for Effective Modeling Constructs," *Business Informatics*, 2016, doi: 10.17323/1998-0663.2016.1.66.79.

[32] M. A. Jaffer, "Can Zakat Charity Help Reduce Economic Inequality?," *International Conference of Zakat*, 2020, doi: 10.37706/iconz.2020.202.

[33] M. Beikirch, S. Cramer, M. J. Frank, P. Otte, E. Pabich, and T. Trimborn, "Simulation of Stylized Facts in Agent-Based Computational Economic Market Models," *arXiv preprint*, 2018, doi: 10.48550/arxiv.1812.02726.

[34] P. Ylikoski, "Social Simulation," in *The Encyclopedia of Social Theory*, 2017, doi: 10.1002/9781118430873.est0759.

[35] T. Trimborn, P. Otte, S. Cramer, M. Beikirch, E. Pabich, and M. Frank, "SABCEMM: A Simulator for Agent-Based Computational Economic Market Models," *Computational Economics*, 2019, doi: 10.1007/s10614-019-09910-1.

[36] S. Kubler, J. Robert, W. Derigent, A. Voisin, and Y. Le Traon, "A state-of-the-art survey & testbed of fuzzy AHP (FAHP) applications," *Expert Systems with Applications*, 2016, doi: 10.1016/j.eswa.2016.08.064.

[37] Y. Li, Y. Zhang, and Y. A. Solangi, "Assessing ESG factors and policies of green finance investment decisions for sustainable development in China using the fuzzy AHP and fuzzy DEMATEL," *Sustainability*, 2023, doi: 10.3390/su152115214.

[38] L. K. Hoang and T. K. Nguyen, "Fuzzy-AHP application in analyzing the factors affecting quality of rural labor," *Journal of Asian Finance, Economics and Business*, 2020, doi: 10.13106/jafeb.2020.vol7.no8.715.

[39] M. Mundzir, R. Zulkarnain, and R. Hardi, "Employing fuzzy AHP in modeling a decision support system for determining scholarship recipients within the university context," *Jurnal Malikussaleh Mengabdi*, 2023, doi: 10.29103/jmm.v2i2.13344.

[40] R.-H. Chiu, L.-H. Lin, and S.-C. Ting, "Evaluation of green port factors and performance: A fuzzy AHP analysis," *Mathematical Problems in Engineering*, 2014, doi: 10.1155/2014/802976.

[41] E. B. Beşikçi, T. Keçeci, Ö. Arslan, and O. Turan, "An application of fuzzy-AHP to ship operational energy efficiency measures," *Ocean Engineering*, 2016, doi: 10.1016/j.oceaneng.2016.05.031.

[42] A. M. Simarmata and Y. Yennimar, "Decision support system for determining land priority for housing development using fuzzy analytical process (Fuzzy-AHP) method," *Sinkron*, 2019, doi: 10.33395/sinkron.v4i1.10243.

[43] H.-Q. Zhang, A. Sekhari, Y. Ouzrout, and A. Bouras, "Optimal inconsistency repairing of pairwise comparison matrices using integrated linear programming and eigenvector methods," *Mathematical Problems in Engineering*, 2014, doi: 10.1155/2014/989726.

A Privacy-Preserving Gaussian Process Regression Framework Against Membership Inference Attacks Using Random Unitary Transformation

Md. Rashedul Islam[1], Jannatul Ferdous Akhi[2], Takayuki Nakachi[3]
Graduate School of Engineering and Science, University of the Ryukyus, Okinawa, Japan[1,2]
Information Technology Center, University of the Ryukyus, Okinawa, Japan[3]

***Abstract*—As artificial intelligence (AI) systems become increasingly embedded in sensitive domains such as healthcare and finance, they face heightened vulnerabilities to privacy threats. A prominent type of attack against AI is the membership inference attack (MIA), which aims to determine whether specific data instances were used in a model's training set, thereby posing a serious risk of sensitive information disclosure. This study focuses on Gaussian Process (GP) models, which are widely adopted for their probabilistic interpretability and ability to quantify predictive uncertainty, and examines their susceptibility to MIAs. To mitigate this threat, a novel defense mechanism based on Random Unitary Transformation (RUT) is introduced, which encrypts training and testing inputs using orthonormal matrices. Unlike Differential Privacy-based Gaussian Processes (DP-GPR), which rely on noise injection and often degrade model performance, the proposed method preserves both the structural integrity and predictive fidelity of the GP model without injecting noise into the learning process. Two configurations are evaluated: i) encryption applied to both training and test data, and ii) encryption applied only to training data. Experimental results on a medical dataset demonstrate that the framework significantly reduces the effectiveness of MIAs while maintaining high predictive accuracy. Comparative analysis with DP-GPR models further confirms that the proposed method achieves competitive or stronger privacy protection with less impact on model utility. These findings underscore the potential of structure-preserving transformations as a practical and effective alternative to noise-based privacy mechanisms in GP models, particularly in privacy-critical machine learning applications.**

***Keywords*—Gaussian process; differential privacy; random unitary transformation; membership inference attack; machine learning**

I. Introduction

Machine learning (ML) is now widely used in many industries due to its popularity and efficacy. However, ethical and legal restrictions restrict the dissemination of private data in sectors such as healthcare and banking. Protecting data privacy is therefore essential at every stage of the ML lifecycle, from developing models to deployment [1]. The widespread use of sensitive data in training neural network models has raised concerns regarding privacy preservation. To assess whether a model inadvertently reveals information about its training data, membership inference attacks (MIAs) [2] have become a widely adopted evaluation technique [3], [4]. In such attacks, adversaries attempt to determine whether a data point was included in the training set by leveraging the model's behavior when presented with that sample. These attacks pose significant risks, especially when inclusion in the training data reveals sensitive information. For instance, if a model is trained using medical imaging data such as MRI scans, disclosing that a particular image was part of the training set could reveal confidential health details. Similarly, training on criminal offender databases may inadvertently expose an individual's criminal background. If an adversary knows a data instance, learning that it was used in training constitutes a breach of confidentiality. MIAs are widely recognized as indicators of privacy risks when ML models are externally accessible. Beyond their standalone impact, MIAs often serve as foundations for advanced attacks such as property inference, where adversaries aim to uncover global attributes of the training data, and profiling attacks, which attempt to infer sensitive characteristics about individuals represented in the data [5].

A widely recognized method in machine learning is the Gaussian Process (GP), which is based on Bayesian nonparametrics [6]. General practitioners can convert linear data into nonlinear formats by integrating domain expertise and knowledge into kernel functions. By fine-tuning the hyperparameters of these kernel functions using Bayes' theorem, it is possible to achieve exceptionally precise estimates. A GP can represent an unlimited number of units in one hidden layer of a neural network [7]. The GP provides both uncertainty (variance) and forecast mean values. By utilizing the uncertainty associated with the GP, it becomes straightforward to assess whether the test data is included in the training data.

A. Related Work

In [1], the authors proposed a privacy technique for GP membership inference to tackle this issue. Differential Privacy-based Gaussian Process (DP-GPR) has become the prevailing framework for ensuring privacy in machine learning [8]. By limiting the influence that any single data point can have on a model's output, DP-GPR provides strong privacy guarantees. This principle has been effectively incorporated into a wide range of machine learning models that assume data point independence, including applications in deep learning, Bayesian regression , and general Bayesian inference techniques such as Markov chain Monte Carlo and variational inference [9], [10], [11], [12]. While (DP-GPR) provides a worst-case guarantee for privacy protection, f-Membership Inference Privacy (f-MIP) leverages noisy stochastic gradient descent (SGD) as a model-agnostic approach to defend against inference attacks. The results show that f-MIP can protect privacy effectively

while using much less noise than DP-GPR, which helps the model keep better performance and accuracy. In contrast, the addition of excessive noise, as typically required by DP-GPR, can reduce model accuracy and disrupt the balance between predictions for training and non-training data. This imbalance may unintentionally create new patterns that adversaries can exploit to infer whether a given test sample was part of the training data. Both DP-GPR and f-MIP defend against inference attacks by injecting noise into the training process. While f-MIP typically uses less noise than DP-GPR and thereby achieves relatively better accuracy, it still introduces some degree of performance degradation due to noise perturbation.

Despite these advances, existing defenses share a fundamental limitation: they depend on noise-based mechanisms that inevitably weaken model utility. Although effective in theory, these approaches reduce accuracy, distort statistical structure, and require careful parameter tuning, making them less practical for real-world deployment. This leaves a clear research gap for defenses that can maintain predictive fidelity while still ensuring strong resistance to MIAs.

B. Contributions of this Study

To address this issue, we propose a secure computation-based defense against MIAs on GP [1]. Our approach uses a RUT to encrypt the data, enabling privacy-preserving processing. By preserving essential geometric properties such as norms, distances, and inner products, the proposed method avoids the trade-off between privacy and accuracy that characterizes noise-based techniques. As a result, attackers cannot determine whether specific test samples were part of the training dataset. Our method addresses the following two scenarios:

1) Case 1: Both the training and test data are encrypted. When the same private key is used for both, the model achieves prediction accuracy comparable to conventional non-encrypted GP regression. In contrast, if different private keys are used (for example, to simulate an attacker scenario), accurate predictions cannot be made, and it becomes infeasible to determine whether test samples were included in the training data.

2) Case 2: Training data is encrypted, while the test data remains non-encrypted. The model fails to make accurate predictions, making it impossible for attackers to determine whether test samples were included in the training set.

The proposed methodology provides a strong defense against membership inference for attackers, while maintaining high prediction accuracy for legitimate users without additional safeguards.

The remainder of this study is organized as follows: Section II introduces the concept of membership inference attacks and their implications for machine learning models. Section III reviews the fundamentals of Gaussian Process Regression. Section IV presents the proposed defense framework against membership inference attacks using Random Unitary Transformation. Section V describes the experimental setup and provides a comparative evaluation of the proposed method. Section VI concludes the study with key findings, while Section VII discusses limitations and outlines directions for future research.

[1]Part of this work has been presented at CSP 2025 [13].

II. Membership Inference Attack

MIAs target ML models with the intent of determining whether a data record was included in the training dataset. These attacks pose serious privacy risks, especially when the mere inclusion of a record reveals sensitive information. For instance, if an attacker learns that a clinical record was used to train a disease-specific model, it may strongly suggest that the individual associated with that record has the disease in question. A recent report by the U.S. National Institute of Standards and Technology (NIST) [14] classifies such inferences as violations of confidentiality. These risks are particularly concerning for organizations offering Machine Learning as a Service (MLaaS), where exposing models to external queries can unintentionally breach legal privacy protections. Veale et al. [15], for example, highlight that MIAs can cause ML outputs to be considered personal data under the General Data Protection Regulation (GDPR) [16].

The idea of MIAs was first introduced by Homer et al. [17] in the genomics domain. They showed that an attacker could use summary statistics from a genomics dataset to find out if a specific person's genome was included. Later, researchers [18], [19] found that similar attacks could also be used on location data. In machine learning, Shokri et al. [2] were the first to show how MIAs could be used against classification models. They found that by only looking at a model's prediction results—without knowing anything about how the model was built—an attacker could tell whether a certain data point was used during training. Since then, extensive research has explored MIAs across various model types, including regression models [20], generative models [21], and embedding models [22]. Alongside this, a growing body of literature has proposed a range of defense mechanisms aimed at mitigating the risk of MIAs while maintaining model performance.

To further illustrate the architecture of a MIA, we present a conceptual overview in Fig. 1. The diagram depicts how an adversary employs a shadow model to approximate the behavior of a target model and then trains an attack model to distinguish between member and non-member data points based on observed prediction outputs. This framework represents the standard MIA strategy widely adopted in the literature and forms the basis of our experimental implementation. The target model is trained using features and corresponding predictions from training data. An adversary queries the target model using features of new data and collects output responses. These outputs, along with corresponding input features, are used to train an attack model (shadow model) that learns to classify whether a data sample was part of the target model's training set (member) or not (non-member).

III. Gaussian Process Regression

A. Gaussian Process

We focus on a regression task where the inputs are vectors $\mathbf{x}_i \in \mathbb{R}^D$ and the outputs are scalar values $y_i \in \mathbb{R}$. Suppose we have a training dataset $D_{\text{train}} = \{\mathbf{X}, \mathbf{Y}\}$, where the input and output matrices are represented, as in Eq. (1):

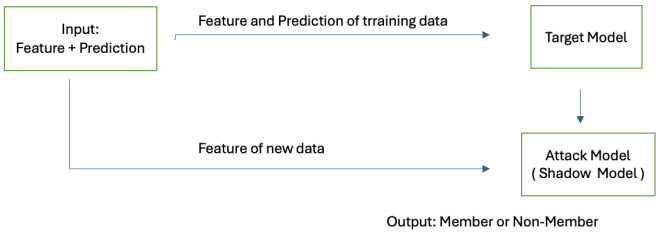


Fig. 1. Architecture of a membership inference attack (MIA) framework.

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_N \end{bmatrix}, \quad \mathbf{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}. \tag{1}$$

The outputs are modeled, as in Eq. (2), where $f(X)$ represents the unknown function to be inferred, and ϵ is Gaussian noise with zero mean and variance σ^2, capturing uncertainty and measurement errors.

$$\mathbf{Y} = f(\mathbf{X}) + \epsilon. \tag{2}$$

This formulation captures measurement noise and reflects uncertainty in the observations. The function $f(\mathbf{X})$ is assumed to follow a GP prior, as expressed in Eq. (3):

$$f(\mathbf{X}) \sim \mathcal{GP}(\mathbf{0},\ \mathbf{K(X,X)}). \tag{3}$$

The mean is set to zero for simplicity, and the covariance between inputs is defined by the kernel $\mathbf{K}$, which is usually a symmetric, positive semi-definite matrix. A commonly used covariance function is the RBF kernel, defined in Eq. (4):

$$\mathbf{K}(\mathbf{x}_i, \mathbf{x}_j) = \theta_1 \exp\left(-\frac{|\mathbf{x}_i - \mathbf{x}_j|^2}{\theta_2}\right) + \sigma^2 \delta(i,j), \tag{4}$$

where, $\delta(i,j)$ 1, if i=j, otherwise 0. To fit the model, the hyperparameters θ_1, θ_2 and σ^2 are optimized by minimizing the negative log marginal likelihood, as shown in Eq. (5):

$$\min_{\mathbf{K},\sigma} \mathbf{Y}^T \left(\mathbf{K(X,\ X)} + \sigma^2 \mathbf{I}\right)^{-1} \mathbf{Y} + \log_2 \left|\mathbf{K(X,\ X)} + \sigma^2 \mathbf{I}\right|. \tag{5}$$

This optimization is typically done via gradient descent [23].

B. Gaussian Process Regression

To make predictions for new input data $\mathbf{x}^* \in \mathbb{R}^D$, we define a test set $D_{\text{test}} = \{\mathbf{X}^*, \mathbf{Y}^*\}$ in Eq. (6):

$$\mathbf{X}^* = \begin{bmatrix} \mathbf{x}_1^* \\ \mathbf{x}_2^* \\ \vdots \\ \mathbf{x}_M^* \end{bmatrix}, \quad \mathbf{Y}^* = \begin{bmatrix} y_1^* \\ y_2^* \\ \vdots \\ y_M^* \end{bmatrix}. \tag{6}$$

The joint distribution of Y and $f(X^*)$ follows a multivariate Gaussian, as expressed in Eq. (7):

$$\begin{bmatrix} \mathbf{Y} \\ f(\mathbf{X}^*) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K(X,\ X)} + \sigma^2 \mathbf{I} & \mathbf{K(X,X^*)} \\ \mathbf{K(X^*, X)} & \mathbf{K(X^*, X^*)} \end{bmatrix}\right). \tag{7}$$

In this formulation, $\mathbf{K(X,X^*)}$ denotes the covariance matrix capturing the dependencies between the N training inputs and the Mtest inputs. Conditioning on the training data, the posterior distribution of $f(X^*)$ is obtained, as in Eq. (8):

$$p(f(\mathbf{X}^*)|(\mathbf{X,Y,X}^*)) \sim \mathcal{N}(f(\mathbf{X}^*), \sigma^2(\mathbf{X}^*)), \tag{8}$$

where, the predictive mean and variance for test inputs are derived using Eq. (9) and Eq. (10), respectively.

$$f(\mathbf{X}^*) = \mathbf{K}(\mathbf{X}^*, \mathbf{X})^T \left[\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathbf{I}\right]^{-1} \mathbf{Y} \tag{9}$$

$$\sigma^2(\mathbf{X}^*) = \mathbf{K}(\mathbf{X}^*, \mathbf{X}^*) - \mathbf{K}(\mathbf{X}^*, \mathbf{X})^T \left[\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathbf{I}\right]^{-1} \mathbf{K}(\mathbf{X}^*, \mathbf{X}). \tag{10}$$

These equations give us both the predicted mean and the uncertainty (variance) for the test data.

IV. Defending Against Gaussian Process Membership Inference Attack

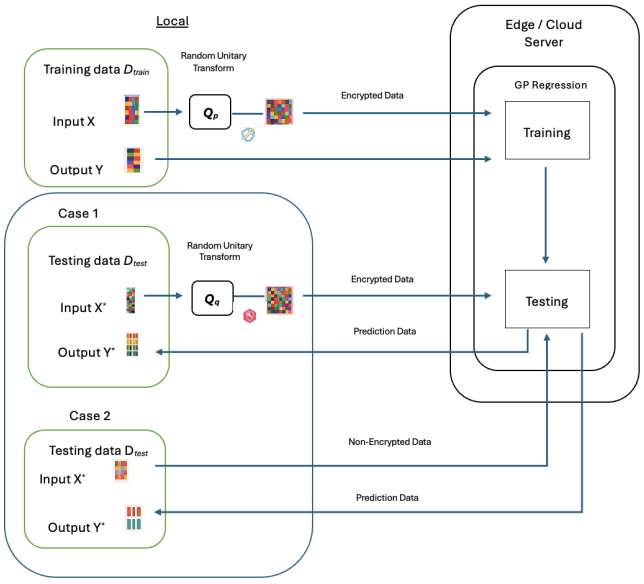


Fig. 2. System configuration of defending against Gaussian process membership inference attack.

A. System Configuration

Fig. 2 illustrates the system architecture designed to defend against MIAs in GP models. The process begins with the collection of training data $D_{\text{train}} = \{\mathbf{X}, \mathbf{Y}\}$, at a local site. To protect the input features $\mathbf{X}$, a random unitary matrix $\mathbf{Q}_p$,

generated using a private encryption key p is applied. This transforms the raw inputs into their encrypted form $\hat{\mathbf{X}}$, which is then transmitted along with the corresponding outputs $\mathbf{Y}$ to an edge or cloud server. At the inference stage, test inputs $\mathbf{X}^*$ are similarly encrypted using another random unitary matrix $\mathbf{Q}_q$, derived from a separate secret key q, resulting in encrypted test inputs $\hat{\mathbf{X}^*}$. These are also sent to the edge or cloud server for processing. This encryption scheme ensures data confidentiality while enabling efficient GP-based inference in distributed environments.

Using the encrypted datasets, the server computes the kernel function and estimates the mean and variance of predictions. When both training and test data are encrypted using the same transformation key (i.e., $p = q$, the predicted results (mean and variance) remain consistent with those produced using non-encrypted data. This property guarantees that encryption does not distort the inference outcomes. In scenarios where the attacker attempts to manipulate the system by inserting non-encrypted test data, the server still processes the data using the encrypted training inputs. The resulting inconsistency in kernel computations between the encrypted and non-encrypted datasets limits the attacker's ability to infer training membership, thereby enhancing the system's privacy protection.

B. Secure Computation

To ensure secure GP regression, the proposed method employs random unitary transformations to encrypt both training and testing data. For the training inputs $\mathbf{X}$, a unitary matrix $\mathbf{Q}_p$ is created using a private key p, resulting in encrypted inputs $\hat{\mathbf{X}} = \mathbf{X}\mathbf{Q}_p$. Similarly, the test inputs $\mathbf{X}^*$ are encrypted as $\hat{\mathbf{X}^*} = \mathbf{X}^*\mathbf{Q}_q$, using a different unitary matrix $\mathbf{Q}_q$ generated with a separate key q, as in Eq. (11):

$$\hat{\mathbf{X}} = \mathbf{X}\mathbf{Q}_p = \begin{bmatrix} \hat{\mathbf{x}}_1 \\ \hat{\mathbf{x}}_2 \\ \vdots \\ \hat{\mathbf{x}}_N \end{bmatrix}, \quad \hat{\mathbf{X}}^* = \mathbf{X}^*\mathbf{Q}_q = \begin{bmatrix} \hat{\mathbf{x}}_1^* \\ \hat{\mathbf{x}}_2^* \\ \vdots \\ \hat{\mathbf{x}}_M^* \end{bmatrix}. \tag{11}$$

This transformation is applied to every element of the input vectors, meaning all input features are protected during transmission and computation. The GP model then estimates the function $f(\hat{\mathbf{X}})$ from the encrypted training data, modeling the outputs, as in Eq. (12):

$$\mathbf{Y} = f(\hat{\mathbf{X}}) + \epsilon. \tag{12}$$

This equation says that the observed output $\mathbf{Y}$ (e.g., a medical measurement) is made up of the model's prediction $f(\hat{\mathbf{X}})$ plus some random noise ϵ to account for uncertainty or measurement error.

Next, we establish a joint Gaussian distribution that encompasses both the training outputs and the model's function values at the encrypted test points, as detailed in Eq. (13):

$$\begin{bmatrix} \mathbf{Y} \\ f(\hat{\mathbf{X}}^*) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \mathbf{K}(\hat{\mathbf{X}}, \hat{\mathbf{X}}) + \sigma^2\mathbf{I} & \mathbf{K}(\hat{\mathbf{X}}, \hat{\mathbf{X}}^*) \\ \mathbf{K}(\hat{\mathbf{X}}^*, \hat{\mathbf{X}}) & \mathbf{K}(\hat{\mathbf{X}}^*, \hat{\mathbf{X}}^*) \end{bmatrix} \right). \tag{13}$$

This joint covariance matrix characterizes a multivariate normal distribution over the training outputs and the function values at the test points. The top-left block $K(\hat{\mathbf{X}}, \hat{\mathbf{X}}) + \sigma^2 I$ captures the dependencies among the encrypted training inputs. The bottom-right $\mathbf{K}(\hat{\mathbf{X}^*}, \hat{\mathbf{X}^*})$, reflects the covariance structure among the encrypted test inputs. The off-diagonal blocks, $\mathbf{K}(\hat{\mathbf{X}}, \hat{\mathbf{X}}^*)$ and $\mathbf{K}(\hat{\mathbf{X}}^*, \hat{\mathbf{X}})$, represent the interaction between the training and test data. The term $\sigma^2 I$ accounts for observation noise, introducing a measure of uncertainty into the model.

From this joint distribution, we derive the posterior prediction—that is, the model's estimate for the test outputs based on the training data:

$$f(\hat{\mathbf{X}}^*) \quad = \quad \mathbf{K}(\hat{\mathbf{X}}^*, \hat{\mathbf{X}})^T \left[\mathbf{K}(\hat{\mathbf{X}}, \hat{\mathbf{X}}) + \sigma^2\mathbf{I} \right]^{-1} \mathbf{Y}. \tag{14}$$

This Eq. (14) is used to generate predictions for the encrypted test inputs $\hat{\mathbf{X}}^*$. It leverages the relationships between test and training inputs, as captured by the kernel function, the internal structure among the training inputs, and the observed training outputs $\mathbf{Y}$.

Eq. (15) provides the variance (uncertainty) of those predictions:

$$\begin{aligned} \sigma^2(\hat{\mathbf{X}}^*) \quad &= \quad \mathbf{K}(\hat{\mathbf{X}}^*, \hat{\mathbf{X}}^*) \\ &- \quad \mathbf{K}(\hat{\mathbf{X}}^*, \hat{\mathbf{X}})^T \left[\mathbf{K}(\hat{\mathbf{X}}, \mathbf{X}) + \sigma^2\mathbf{I} \right]^{-1} \mathbf{K}(\hat{\mathbf{X}}^*, \hat{\mathbf{X}}). \end{aligned} \tag{15}$$

This expression quantifies the model's predictive uncertainty for each test input. A lower variance indicates higher confidence in the prediction, whereas a higher variance suggests greater uncertainty. The magnitude of this variance reflects how closely the test input resembles the training data distribution.

C. Encryption Based on Random Unitary Transform

Prior research has investigated secure sparse coding techniques based on random unitary transformation [24], [25], [26], [27], demonstrating that it is possible to protect sensitive data while preserving key geometric structures. To ensure data privacy in Gaussian Process Regression (GPR), this study adopts a transformation-based encryption mechanism utilizing random unitary matrices. This technique obfuscates input data while preserving essential geometric properties required for kernel-based learning. Unlike additive noise methods such as DP-GPR, this approach enables secure learning without degrading model accuracy.

Let $\mathbf{x}_i^* \in \mathbb{R}^D$ denote a raw input vector. A unitary matrix $\mathbf{Q}_p \in \mathbb{C}^{DXD}$, generated using a private key p, is applied to encrypt the data, as shown in Eq. (16):

$$\hat{\mathbf{x}_i} = \mathbf{x}_i \mathbf{Q}_p. \tag{16}$$

This transformation yields an encrypted representation $\hat{\mathbf{x}}_i$, where $\mathbf{Q}_p$ satisfies the unitary condition given in Eq. (17):

$$\mathbf{Q}_p^H \mathbf{Q}_p = \mathbf{I}. \tag{17}$$

Here, $\mathbf{Q}_P^H$ denotes the Hermitian transpose, and $\mathbf{I}$ is the identity matrix. This property ensures that the transformation is norm-preserving and invertible, enabling the encrypted inputs to retain structural similarity to the original data. The encryption preserves the following fundamental properties:

- Property 1: Norm Isometry

$$||\mathbf{x}_\mathrm{i}||_2^2 = ||\widehat{\mathbf{x}}_\mathrm{i}||_2^2$$

- Property 2: Distance Preservation

$$||\mathbf{x}_\mathrm{i} - \mathbf{x}_\mathrm{j}||_2^2 = ||\widehat{\mathbf{x}}_\mathrm{i} - \widehat{\mathbf{x}}_\mathrm{j}||_2^2$$

- Property 3: Inner Product Preservation

$$\mathbf{x}_i^H \mathbf{x}_\mathrm{j} = \widehat{\mathbf{x}}_i^H \widehat{\mathbf{x}}_j$$

These preserved characteristics are essential for accurate kernel computation in GPR, particularly when using the Radial Basis Function (RBF) kernel. The kernel function on the encrypted data is expressed, as in Eq. (18):

$$K(\hat{\mathbf{x}_i}, \hat{\mathbf{x}_j}) = \theta_1 \exp\left(-\frac{||\hat{\mathbf{x}_i} - \hat{\mathbf{x}_j}||^2}{\theta_2}\right) + \sigma^2 \delta(i,j). \tag{18}$$

Substituting $\hat{\mathbf{x}}_i = \mathbf{x}_i \mathbf{Q}_p$ and $\hat{\mathbf{x}}_j = \mathbf{x}_j \mathbf{Q}_p$, and applying the unitary property, we obtain the distance preservation relation shown in Eq. (19):

$$||\hat{\mathbf{x}_i} - \hat{\mathbf{x}_j}||^2 = ||(\mathbf{x}_i - \mathbf{x}_j)\mathbf{Q}_p||^2 = ||(\mathbf{x}_i - \mathbf{x}_j)||^2. \tag{19}$$

Hence, the kernel function becomes: $K(\hat{\mathbf{x}_i}, \hat{\mathbf{x}_j})$ = $K(\mathbf{x}_i, \mathbf{x}_j)$ This result confirms that encrypting the input vectors with the same key does not alter the pairwise kernel values, thereby preserving model fidelity during training.

While the analysis primarily focuses on the RBF kernel, the properties of random unitary transformation (RUT)—such as norm preservation, inner product invariance, and distance preservation—also ensure compatibility with other kernel functions. For example, the linear kernel, which directly relies on inner products, remains unaffected by RUT. Similarly, polynomial kernels, which are functions of inner products, also retain their behavior under unitary transformations, enabling broad applicability of the proposed method across kernel-based models.

Similarly, if the test inputs $\mathbf{x}_i^* \in \mathbb{R}^D$ are encrypted with the same unitary matrix $\mathbf{Q}_p$, the kernel between test points remains invariant, as expressed as in Eq. (20):

$$K(\hat{\mathbf{x}_i^*}, \hat{\mathbf{x}_j^*}) = K(\mathbf{x}_i^*, \mathbf{x}_j^*). \tag{20}$$

However, when training and test inputs are encrypted using different keys (i.e., $\mathbf{Q}_p \neq \mathbf{Q}_q$), the kernel function between them is modified, as in Eq. (21):

$$K(\hat{\mathbf{x}_i}, \hat{\mathbf{x}_j}) = \theta_1 \exp\left(-\frac{||\hat{\mathbf{x}_i}\mathbf{Q}_q - \hat{\mathbf{x}_j}\mathbf{Q}_p||^2}{\theta_2}\right) + \sigma^2 \delta(i,j). \tag{21}$$

Since $\mathbf{Q}_p \neq \mathbf{Q}_q$, the transformation no longer preserves the distances between training and test samples. Consequently, $K(\hat{\mathbf{x}_i^*}, \hat{\mathbf{x}_j}) \neq K(\mathbf{x}_i, \mathbf{x}_j)$. This mismatch distorts the kernel values, making it infeasible for an adversary to infer training membership based on similarity, thereby enhancing the model's privacy guarantees.

When the same transformation key is applied to both training and test inputs (that is, $\mathbf{Q}_p = \mathbf{Q}_q$), the encrypted GP model produces predictions and uncertainties identical to the plaintext case, as shown in Eq. (22) and Eq. (23):

$$f(\widehat{\mathbf{X}}^*) = f(\mathbf{X}^*), \tag{22}$$

$$\sigma^2(\widehat{\mathbf{X}}^*) = \sigma^2(\mathbf{X}^*). \tag{23}$$

This property ensures that the encryption does not compromise the accuracy of the model.

Finally, when only the training inputs are encrypted (i.e., test data remains non-encrypted), the kernel between test and training data becomes distorted, as expressed in Eq. (24):

$$\begin{aligned} K(\mathbf{x}_\mathrm{i}^*, \widehat{\mathbf{x}}_\mathrm{j}) &= \theta_1 \exp\left(-\frac{|\mathbf{x}_\mathrm{i}^* - \widehat{\mathbf{x}}_\mathrm{j}|^2}{\theta_2}\right) + \sigma^2 \delta(i,j) \\ &= \theta_1 \exp\left(-\frac{|\mathbf{x}_i\mathbf{I} - \mathbf{x}_j\mathbf{Q}_p|^2}{\theta_2}\right) + \sigma^2 \delta(i,j) \\ &\neq K(\mathbf{x}_i, \mathbf{x}_j). \end{aligned} \tag{24}$$

Since the test inputs are in their original form and the training inputs are encrypted, the kernel function again deviates from its non-encrypted counterpart, further reinforcing privacy against inference attacks.

V. Experimental Evaluations

To evaluate the effectiveness of the proposed approach, a series of experiments were conducted using a diabetes dataset commonly employed in medical data analysis.

A. Simulation Conditions

To assess the effectiveness of the proposed secure Gaussian Process Regression (secGPR) framework, experiments were conducted using a publicly available medical dataset related to diabetes. This dataset, sourced from the scikit-learn library, comprises 442 individual records, each containing 10 baseline clinical features such as age, sex, body mass index (BMI), and a quantitative measure of disease progression recorded one year after baseline assessment [28], [29]. For the purpose of model training, a subset of 353 samples was used, while the remaining 89 records were reserved for testing. In this

experimental setup, the input matrix $\mathbf{X} \in \mathbb{R}^{NXD}$ represents the 10-dimensional clinical features, and the corresponding target values $\mathbf{Y} \in \mathbb{R}^{N}$ indicate the diabetes progression scores. This configuration allows for a rigorous evaluation of the secGPR model's ability to accurately predict continuous outcomes under various encryption scenarios.

The random unitary transformation matrix $\mathbf{Q}_p$, used to encrypt the input features, was generated through the Gram–Schmidt orthogonalization process to ensure it satisfies the unitary constraint. This matrix was applied to the input data prior to training and testing in the secure learning environment.

B. Simulation Results

This section presents a comparative analysis of the predictive performance between the conventional GP model and the proposed secure approach.

1) Non-encrypted GP (Where both the training data $\boldsymbol{X}$ and the testing data $\boldsymbol{X}^*$ remain non-encrypted): When the test data was included in the training set, the conventional (non-encrypted) GP model exhibited near-perfect predictive performance, achieving an average error of 1.515×10^{-11} and 1.000×10^{-10}, as reported in Table I. While these results indicate extremely high accuracy, they also reveal a significant security concern: the model's highly deterministic behavior enables adversaries to easily infer whether specific data points were used during training, thus exposing it to MIAs.

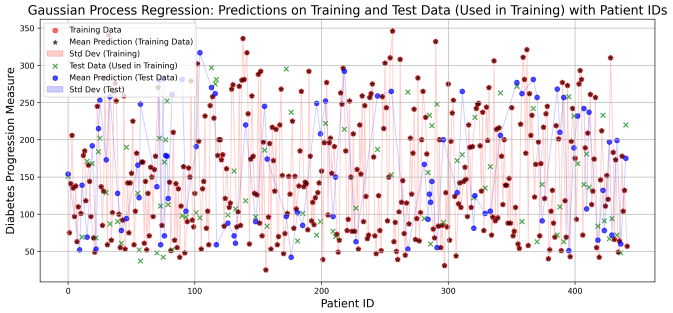


Fig. 3. Mean prediction and uncertainty of conventional GP for non-encrypted training and testing data.

Conversely, when the test data was excluded from the training process, the model's predictive accuracy declined markedly, with the average error rising to 64.32 and the variance increasing to 585.26. This outcome reflects a more realistic deployment scenario in which unseen inputs yield greater uncertainty and lower confidence. The sharp contrast between these two settings highlights a fundamental vulnerability of traditional GPR models in privacy-sensitive applications: they lack an effective balance between predictive performance and data confidentiality.

Fig. 3 illustrates the prediction results of the GPR model in the non-encrypted baseline configuration, where both training and test data are used in their original, untransformed form. The x-axis represents patient IDs, and the y-axis corresponds to the diabetes progression measure.

In this figure:

- Red circles denote the ground truth values of the training data.
- Brown stars represent the model's mean predictions for training samples, with the red shaded region indicating the corresponding predictive uncertainty (standard deviation).
- Green crosses mark the test data samples that were included in the training set.
- Blue circles show the predicted means for these test samples, while the blue shaded region visualizes their associated uncertainty.

The predictions for both training and test data exhibit excellent alignment with the true values, accompanied by consistently low variance. This high confidence and low prediction error—particularly for test data reused during training—indicates strong memorization behavior. While desirable from a model accuracy standpoint, this deterministic prediction pattern introduces a substantial privacy vulnerability. Specifically, the marked discrepancy in model behavior between training and unseen test inputs (not shown in this figure) can be exploited in MIAs. In this setting, an adversary can reliably infer whether a data point was part of the training set by observing prediction confidence and error.

TABLE I. COMPARATIVE PERFORMANCE ACROSS METHODS

Scenario	Condition	Average Error	Average Variance
Non-encrypted GP	Test Data Included	1.515×10^{-11}	1.000×10^{-10}
	Test Data Excluded	64.32	585.26
DP-GPR	Test Data Included	45.7158	342.4383
	Test Data Excluded	50.2886	345.6273
Proposed GP: Case 1 ($P = Q$)	Test Data Included	1.515×10^{-11}	1.000×10^{-10}
	Test Data Excluded	64.32	585.26
Proposed GP: Case 1 ($P \neq Q$)	Test Data Included	102.604	838.253
	Test Data Excluded	87.07	850.8863
Proposed GP: Case 2	Test Data Included	113.96	843.99
	Test Data Excluded	103.74	827.38

2) DP-GPR: The DP-GPR based model achieves moderate privacy protection by injecting noise into the training process. It achieves relatively lower error and variance compared to Case 2 (included: error = 45.7158, variance = 342.4383; excluded: error = 50.2886, variance = 345.6273), while still maintaining acceptable accuracy. However, the residual gap between seen and unseen data behaviors indicates that some MIA risk remains. Moreover, DP-GPR distorts the internal structure of the data and relies heavily on tuning privacy budgets, potentially impacting generalization.

Fig. 4 illustrates the predictive performance of the DP-GPR model when the test data is included in the training set. The horizontal axis represents the individual patient IDs, while the vertical axis corresponds to the diabetes progression measure. The red dots denote the true test data values, serving as the ground truth. The blue line with star markers represents the mean predictions made by the DP-GPR model. The pink shaded region around the prediction curve indicates the standard deviation of the predictions, capturing the model's predictive uncertainty. As shown, the prediction curve closely follows the general trend of the true values, but several true points deviate significantly—especially in regions of higher variance.

These deviations are partially accounted for by the uncertainty bands, which widen in areas where the model expresses less confidence. The presence of these variability bands is a result of the noise added by the DP-GPR mechanism, which affects both prediction sharpness and confidence calibration.

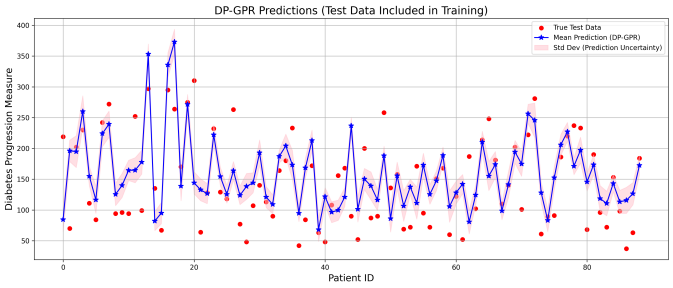


Fig. 4. Mean prediction and uncertainty of DP-GPR.

Fig. 4 highlights a key characteristic of DP-GPR: while the model maintains reasonable alignment with the true data, it introduces a moderate level of uncertainty due to privacy-preserving noise. Importantly, the overall prediction behavior remains consistent, though less precise compared to non-private or fully encrypted models, such as those in the proposed method's Case 1.

3) Case 1 (Both the training data $\boldsymbol{X}$ and the test data $\boldsymbol{X}^*$ were encrypted using random unitary transformations): Two scenarios were evaluated to assess the effectiveness of the proposed method: one in which the same key was used for both transformations ($\mathbf{Q}_p = \mathbf{Q}_q$), and another in which different keys were applied ($\mathbf{Q}_p \neq \mathbf{Q}_q$).

In the key mismatch scenario ($\mathbf{Q}_p \neq \mathbf{Q}_q$), the model's predictive performance declined due to the inconsistency in the transformed input spaces. When test data was included in the training set, the average prediction error and variance were 102.60 and 838.25, respectively. When test data was excluded, the error reduced to 87.07, accompanied by a slight increase in variance to 850.89. These results indicate that although encryption mismatches degrade accuracy, they also enhance privacy by disrupting kernel-based similarity, thereby complicating membership inference attempts.

In contrast, the key match scenario ($\mathbf{Q}_p = \mathbf{Q}_q$) yielded prediction results indistinguishable from those of the non-encrypted model. Specifically, when test data was part of the training set, the average error and variance were extremely low— 1.515×10^{-11} and 1.000×10^{-10}, respectively. Even when the test data was excluded, the model maintained strong performance with an error of 64.32 and variance of 585.26. These findings demonstrate that when consistent encryption is applied, the proposed method retains full predictive accuracy while ensuring data confidentiality.

Fig. 5 presents the predictive results of the proposed GPR model under the Case 1 configuration, where both training and test inputs are encrypted using the same random unitary matrix (i.e., $\mathbf{Q}_p = \mathbf{Q}_q$). The x-axis represents patient IDs, while the y-axis denotes diabetes progression measurements.

In the plot, red dots correspond to the true training data, with black stars showing the model's predicted means and red

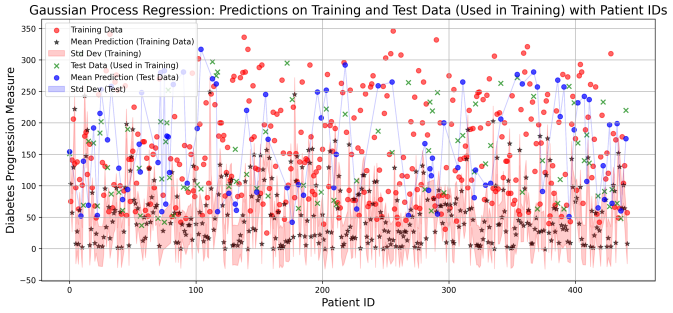


Fig. 5. Mean prediction and uncertainty of proposed method (Case 1) for encrypted training and testing data.

shaded areas indicating prediction uncertainty. Green crosses represent the encrypted test data (used in training), while blue circles and blue shaded regions indicate their predicted means and standard deviations, respectively.

The figure shows that the model achieves highly accurate predictions and low uncertainty for both datasets, indicating that structural data relationships are preserved under consistent encryption.

4) Case 2 (Where training data $\boldsymbol{X}$ is encrypted and testing data $\boldsymbol{X}^*$ is non-encrypted): This asymmetric configuration, referred to as a mixed encryption state, resulted in the poorest predictive performance among all evaluated scenarios. When the test data was included in the training set, the model exhibited the highest average error and variance, recorded at 113.96 and 844.00, respectively (see Table I). Excluding the test data from training slightly improved the outcomes, with the average error decreasing to 103.74 and the variance dropping to 827.38. Despite this modest reduction, the performance remained significantly worse than in fully encrypted or non-encrypted settings.

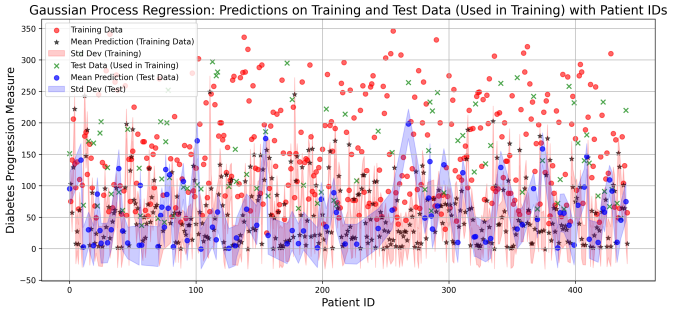


Fig. 6. Mean prediction and uncertainty of proposed method (Case 2) for encrypted training data and non-encrypted testing data.

Fig. 6 illustrates the prediction performance of the proposed GPR model under the Case 2 configuration, in which the training data is encrypted using a random unitary matrix $\mathbf{Q}_p$, while the test data remains non-encrypted. The x-axis represents the patient IDs, and the y-axis indicates the diabetes progression measure.

In this figure:

- Red circles correspond to the ground truth values for the training data.

- Black stars represent the mean predictions for the training data, and the red shaded area denotes the associated prediction uncertainty (standard deviation).
- Green crosses mark the true test data values (used in training).
- Blue circles indicate the predicted mean values for the test data, with the blue shaded area visualizing their predictive uncertainty.

Despite the fact that test data are included in training, the model demonstrates substantially degraded prediction accuracy and elevated uncertainty across both training and test datasets. This performance degradation arises from the incompatibility between the encrypted training data and the non-encrypted (or differently encrypted) test data, which undermines kernel consistency and limits the model's ability to capture meaningful relationships.

These results indicate that the mixed encryption scenario in Case 2 provides a high level of privacy by intentionally disrupting the correspondence between encrypted and non-encrypted data. The resulting reduction in predictive accuracy is not a drawback, but rather a desirable characteristic from a privacy perspective—specifically, it ensures that attackers cannot make accurate estimations about whether a given sample was part of the training set. The sharp degradation in model performance serves to obscure membership signals, thereby strengthening the model's resistance to membership inference attacks (MIAs).

VI. Conclusion

This study introduced a noise-free privacy-preserving framework for Gaussian Process Regression based on Random Unitary Transformation. The method supports configurations that either preserve accuracy (Case 1) or prioritize privacy robustness (Case 2). Compared with DP-GPR, which relies on noise injection, the proposed framework maintains model fidelity while enhancing resistance to MIAs.

The results demonstrate that Case 1 achieves excellent predictive accuracy, comparable to the non-encrypted baseline, while ensuring structural privacy without introducing noise. However, the observed discrepancy in prediction behavior between training and unseen data introduces a moderate vulnerability to MIAs. In contrast, Case 2 offers the strongest privacy protection, as it produces uniformly high error and uncertainty regardless of whether a data point was part of the training set. This consistency effectively conceals membership status, though at the cost of reduced model utility.

Compared to DP-GPR, which achieves moderate privacy through noise injection, the proposed framework provides a noise-free alternative that either maintains model fidelity or maximizes privacy robustness. While DP-GPR strikes a balance between accuracy and privacy, it fails to eliminate membership leakage entirely and distorts the data's statistical structure.

Overall, the proposed approach demonstrates flexibility in supporting different privacy scenarios without compromising predictive accuracy. In Case 1, where the same key is used to encrypt both training and test data, the model maintains predictive performance equivalent to the non-encrypted baseline. However, this configuration introduces a potential attack surface: if the shared key is ever leaked, inferred, or reused, it could allow adversaries to align encrypted inputs and compromise membership privacy. In contrast, Case 2, where only the training data is encrypted, offers stronger privacy protection by fully disrupting alignment between training and test distributions. The resulting decrease in predictive accuracy is not a limitation, but a privacy-enhancing effect—by weakening the correspondence between training and test samples, it becomes significantly more difficult for adversaries to infer membership or reconstruct sensitive data. These findings highlight the promise of unitary transformation-based encryption as a practical, interpretable, and privacy-preserving solution for sensitive machine learning applications.

VII. Limitations and Future Work

While the proposed encryption-based GPR framework demonstrates strong potential in balancing privacy and predictive performance, several limitations warrant further investigation. First, the current evaluation is conducted on a single dataset with a moderate number of features and samples. Additional experiments on larger, more diverse datasets—particularly in high-dimensional or real-time environments—are necessary to assess the scalability and generalizability of the approach. Second, although Case 2 offers strong resistance to MIAs, it does so at the cost of significantly degraded predictive accuracy, which may limit its practicality in applications where precision is critical. Furthermore, this work assumes a passive attacker model; evaluating robustness under more aggressive or adaptive adversarial strategies (e.g., shadow models, reconstruction attacks) remains an open area.

Future research could explore hybrid defense mechanisms that combine unitary transformation with adaptive noise injection or output randomization to further enhance privacy without significantly compromising predictive accuracy. Investigating how this framework performs under federated learning or distributed settings is also a promising direction—particularly in scenarios where local privacy constraints and communication efficiency are critical. These directions would broaden the applicability of the proposed method and address emerging challenges in real-world privacy-preserving machine learning systems.

References

[1] T. Leemann, M. Pawelczyk, and G. Kasneci, "Gaussian membership inference privacy," Advances in Neural Information Processing Systems, 36, pp.73866-73878, 2024.

[2] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," In 2017 IEEE symposium on security and privacy (SP) (pp. 3-18), 2017.

[3] S. K. Murakonda, and R. Shokri, "Ml privacy meter: Aiding regulatory compliance by quantifying the privacy risks of machine learning," arXiv preprint arXiv:2007.09339, 2020.

[4] S. Song, and D. Marn, "Introducing a new privacy testing library in tensorflow," URL https://blog. tensorflow. org/2020/06/introducing-new-privacy-testing-library. html, 2020.

[5] E. De Cristofaro, "An overview of privacy in machine learning," arXiv preprint arXiv:2005.08679, 2020.

[6] H. Liu, Y. S. Ong, X. Shen, and J. Cai, "When Gaussian process meets big data: A review of scalable GPs," IEEE Transactions on neural networks and learning systems, 31(11), 4405-4423, 2020.

[7] R. M. Neal, "Bayesian learning for neural networks," (Vol. 118). Springer Science & Business Media, 2012.

[8] C. Dwork, F. McSherry, K. Nissim and A. Smith, "Calibrating noise to sensitivity in private data analysis," In Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006. Proceedings 3 (pp. 265-284). Springer Berlin Heidelberg, 2006.

[9] M. Abadi, A. Chu, I. Goodfellow, H.B. McMahan, I. Mironov, K. Talwar and L. Zhang, October. "Deep learning with differential privacy," In Proceedings of the 2016 ACM SIGSAC conference on computer and communications security (pp. 308-318), 2016.

[10] G. Bernstein, and D.R. Sheldon, "Differentially private bayesian linear regression," Advances in Neural Information Processing Systems, 32, 2019.

[11] M. Heikkilä, J. Jälkö, O. Dikmen and A. Honkela, "Differentially private markov chain monte carlo," Advances in Neural Information Processing Systems, 32, 2019.

[12] J. Jälkö, O. Dikmen and A. Honkela, "Differentially private variational inference for non-conjugate models," arXiv preprint arXiv:1610.08749, 2016.

[13] M. R. Islam, J. F. Akhi, and T. Nakachi, "Defending Against Gaussian Process Membership Inference Attack," presented at the *IEEE Conf. on Cyber Security and Privacy (CSP)*, Okinawa, Japan, 2025.

[14] E. Tabassi, K.J. Burns, M. Hadjimichael, A.D. Molina-Markham and J.T. Sexton, "A taxonomy and terminology of adversarial machine learning," NIST IR, pp.1-29, 2019.

[15] M. Veale, R. Binns, and L. Edwards, "Algorithms that remember: model inversion attacks and data protection law," Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 376(2133), p.20180083, 2018.

[16] European Parliament and Council of the European Union, "Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)," Official Journal of the European Union, vol. L 119, pp. 1–88, May 2016.

[17] N. Homer, S. Szelinger, M. Redman, D. Duggan, W. Tembe, J. Muehling, J.V. Pearson, D.A. Stephan, S.F. Nelson and D.W. Craig, "Resolving individuals contributing trace amounts of DNA to highly complex mixtures using high-density SNP genotyping microarrays," PLoS genetics, 4(8), p.e1000167, 2008.

[18] A. Pyrgelis, C. Troncoso and E. De Cristofaro, "Knock knock, who's there? Membership inference on aggregate location data," arXiv preprint arXiv:1708.06145, 2018.

[19] A. Pyrgelis, C. Troncoso, and E. De Cristofaro, "Measuring membership privacy on aggregate location time-series," Proceedings of the ACM on Measurement and Analysis of Computing Systems, 4(2), pp.1-28, 2020.

[20] U. Gupta, D. Stripelis, P.K. Lam, P. Thompson, J.L. Ambite and G. Ver Steeg, "Membership inference attacks on deep regression models for neuroimaging," In Medical Imaging with Deep Learning (pp. 228-251). PMLR, 2021.

[21] J. Hayes, L. Melis, G. Danezis and E.L. De Cristofaro, "Membership Inference Attacks Against Generative Models"; URL https://api. semanticscholar. org/CorpusID, 202588705, 2018.

[22] C. Song and A. Raghunathan, "Information leakage in embedding models," In Proceedings of the 2020 ACM SIGSAC conference on computer and communications security (pp. 377-390), 2020.

[23] C. K. Williams, and C. E. Rasmussen, "Gaussian processes for machine learning," (Vol. 2, No. 3, p. 4). Cambridge, MA: MIT press, 2006.

[24] T. Nakachi, Y. Bandoh, and H. Kiya, "Secure overcomplete dictionary learning for sparse representation," IEICE Transactions on Information and Systems, 103(1), 50-58, 2020.

[25] T. Nakachi, and H. Kiya, "Secure OMP computation maintaining sparse representations and its application to EtC systems," IEICE Transactions on Information and Systems, 103(9), 1988-1997, 2020.

[26] Y. Wang, and T. Nakachi, "A privacy-preserving learning framework for face recognition in edge and cloud networks," IEEE Access, 8, 136056-136070, 2020.

[27] Y. Bandoh, T. Nakachi, and H. Kiya, "Distributed secure sparse modeling based on random unitary transform," IEEE Access, 8, 211762-211772, 2020.

[28] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," The Annals of Statistics, vol. 32, no. 2, pp. 407–499, 2004

[29] scikit-learn, https://scikit-learn.org/stable/index.html.

Autonomous Driving in Adverse Weather: A Multi-Modal Fusion Framework with Uncertainty-Aware Learning for Robust Obstacle Detection

Zhengqing Li[1], Baljit Singh Bhathal Singh[2]
College of Engineering, Universiti Teknologi MARA, 40450 Shah Alam, Malaysia[1, 2]
School of New Energy and Intelligent Networked Automobile, University of Sanya, 572022, Sanya, Hainan[1]
New Energy and Intelligent Vehicle Engineering Research Center of Hainan Province, 572022, Sanya, Hainan[1]

***Abstract*—Robust obstacle detection in autonomous driving under adverse weather remains a critical challenge due to sensor degradation, visibility reduction, and increased uncertainty. This study proposes an Uncertainty-Aware Multi-Modal Fusion (UAMF) framework that integrates LiDAR, RGB images, and weather priors through a dynamic cross-modal attention mechanism and Bayesian uncertainty modeling. The model adaptively adjusts the fusion weights between sensor modalities according to real-time weather conditions and jointly optimizes detection loss with a KL divergence regularization to quantify predictive uncertainty. Experimental results on the nuScenes, KITTI-Adverse, and CARLA datasets demonstrate that UAMF achieves superior performance across rain, snow, and fog scenarios, with mAP@0.5 reaching 0.78, 0.72, and 0.65, respectively—representing 12–31% gains over existing baselines. Notably, UAMF reduces false positive rates by up to 40% in low-visibility conditions and exhibits a strong correlation ($\rho = 0.85$) between estimated uncertainty and localization error. Ablation studies confirm the importance of the weather-aware fusion and uncertainty modules, while visibility-level analysis shows improved robustness under <30 m scenarios. The proposed framework offers reliable uncertainty signals for downstream decision-making and is deployable in real-time on embedded platforms. Future work will explore unsupervised weather parameter estimation, uncertainty-aware trajectory forecasting, and cross-domain generalization.**

Keywords*—Autonomous driving; adverse weather; multimodal sensor fusion; Bayesian neural networks; uncertainty estimation

I. Introduction

In recent years, autonomous driving technology has made great progress under good weather conditions. However, in severe weather environments such as heavy rain, heavy snowfall and dense fog, the stability and reliability of autonomous driving perception systems still face major challenges. According to Fang et al. (2022), about 30% of perception system failure cases are directly related to weather factors, manifested in problems such as sensor signal attenuation, obstacle occlusion and reduced environmental visibility. For example, LiDAR sensors have sparse data due to point cloud scattering in rainy and snowy weather, which seriously affects the accuracy of target detection [1]; at the same time, cameras are prone to miss key targets in haze weather due to reduced image contrast [2]. Although current research generally adopts multimodal sensor fusion (such as LiDAR, camera and millimeter wave radar) to improve system robustness [3], existing methods still have significant shortcomings in practical applications. Specifically, the first is the lack of effective modeling of the time-varying characteristics of multimodal data under severe weather conditions; the second is the lack of reasonable quantification of potential uncertainties in the detection results, resulting in a high false detection rate in low visibility environments.

In the perception module of the autonomous driving system, severe weather conditions remain a key problem that affects the accuracy and robustness of environmental perception. Existing research mainly conducts modeling and training under ideal or slightly disturbed meteorological conditions, resulting in a significant decrease in the performance of the perception system under conditions such as heavy rain, heavy snow or dense fog. The main problems include: multimodal sensor data finds it difficult to achieve an efficient alignment under dynamic weather conditions, and the accuracy of feature fusion is limited; detection models generally lack uncertainty modeling and cannot effectively identify false detections caused by perceptual ambiguity; in addition, data resources under real severe weather are extremely limited. Although simulation data can make up for some sample gaps, there is a significant domain difference between it and the real environment, which restricts the generalization ability of the model in actual scenarios.

This study aims to address the above problems and propose a weather-aware multimodal detection framework that integrates uncertainty modeling to improve the robustness and adaptability of autonomous driving systems in complex weather environments. Specifically, this study will construct a cross-modal fusion mechanism guided by weather parameters to achieve effective alignment of sensor features under dynamic meteorological conditions; introduce a Bayesian neural network structure to model the uncertainty of target detection results, thereby effectively suppressing false detection and missed detection in low-visibility scenarios; at the same time, construct a joint training strategy that combines simulation and real data, and use domain adaptation technology to alleviate the distribution offset between training and application scenarios, and improve the generalization performance of the model in actual deployment environments.

The contributions of this study are mainly reflected in three aspects: first, weather intensity is introduced as prior information in the multimodal fusion strategy for the first time, which effectively improves the accuracy and stability of sensor data alignment under severe weather conditions; second, combined with the Bayesian inference mechanism, end-to-end modeling of the confidence of the detection results is realized, providing the system with a decision-making basis based on uncertainty perception; finally, by constructing a collaborative training framework of simulation and real data and adopting progressive fine-tuning and domain randomization methods, the cross-domain generalization ability of the model in different weather scenarios is significantly improved, enhancing its practicality and robustness in real complex environments.

II. Literature Review

In recent years, academic research on the problem of target detection in autonomous driving systems under severe weather conditions has mainly focused on two technical paths: enhanced optimization of a single modality and deep fusion of multimodal information. In response to the performance degradation of specific sensors in extreme weather conditions, researchers have proposed a series of image and point cloud enhancement methods. For example, [5] constructed the FogNet model, which combined the physically driven image defogging mechanism with the end-to-end joint optimization of the detection task, and significantly improved the target detection accuracy under dense fog conditions. In terms of LiDAR, [4] designed the DETR-Weather model using the deformable Transformer structure, and alleviated the occlusion interference problem in heavy rain scenes through the dynamic attention mechanism. Although such methods have achieved certain results in a single sensor channel, due to the inherent problems of high camera noise in low light and sparse LiDAR point clouds on snowy days, it is difficult to completely avoid them, resulting in insufficient performance in actual scenes where multiple meteorological interferences overlap.

In contrast, multimodal fusion methods have shown greater potential in improving system robustness because they can integrate multi-source sensor information. Fusion strategies mainly include early fusion, late fusion, and feature-level fusion. Researchers have conducted extensive explorations around the timing and mechanism of fusion. After systematically comparing different fusion strategies, [2] pointed out that early fusion in rainy and foggy environments results in performance degradation due to the lack of alignment of features between modalities, and suggested introducing a mid- and late-stage feature alignment mechanism to enhance robustness. In [3], the authors proposed a fusion model based on a generative adversarial network (GAN), which expanded the training set by synthesizing severe weather data and improved the detection performance to a certain extent. However, this method does not explicitly introduce dynamic modeling of weather parameters, resulting in significant domain differences between training data and real scenes [6], which affects the actual deployment effect of the model.

With the increasing requirements for the stability of detection systems, uncertainty modeling has gradually become a key direction for autonomous driving perception research in severe weather environments. At present, related methods are mainly based on Bayesian deep learning and probabilistic modeling frameworks. The Bayesian convolutional neural network proposed by [8] uses Monte Carlo sampling to estimate the uncertainty of the prediction distribution, laying a theoretical foundation for subsequent research. However, its high computational cost limits its applicability in real-time scenarios. To improve efficiency, [9] introduced a lightweight Bayesian detection head and implemented joint training with three-dimensional object detection, effectively reducing the false detection rate while maintaining real-time performance. In addition, [7] proposed a multi-task uncertainty weighting mechanism based on heteroscedasticity modeling to achieve dynamic loss balance between classification and regression tasks, but did not perform specific modeling for weather disturbances. In [4], the authors further incorporated weather variables into detection error modeling and constructed a more adaptive unimodal detection model through conditional variational autoencoders (CVAEs), but their method has not yet been extended to multimodal scenarios.

Although existing research has made some progress in improving the performance of perception systems, from a comprehensive perspective, current methods still have many limitations. First, multimodal fusion models generally do not embed weather parameters as dynamic priors into the fusion mechanism, resulting in unstable feature alignment strategies under rapidly changing meteorological conditions such as rain, snow, and fog [2] [3]. Second, current uncertainty modeling mostly focuses on the model prediction distribution itself and has not yet systematically considered the external environmental uncertainty introduced by weather disturbances. This neglect makes it difficult for the model to distinguish between sensor noise and model entity errors [5] [7]. Finally, although simulation platforms (such as CARLA) can generate diverse severe weather data to alleviate the problem of sample scarcity, due to the significant distribution offset between their synthetic features and real data, directly training models often experiences performance degradation in actual tests. In [6], the authors pointed out that the current mainstream synthetic training strategy still suffers from a performance degradation of more than 25% when deployed in real scenarios. There is an urgent need to build a more targeted simulation-real data collaborative training and domain adaptation mechanism to improve the model's cross-scenario generalization ability.

III. Methodology

A. Model Framework

This study proposes an uncertainty-aware multi-modal fusion framework (UAMF), whose core architecture consists of four modules: multimodal feature extraction, weather-aware cross-modal fusion, Bayesian uncertainty modeling, and target detection head. The multimodal feature extraction module is used to process LiDAR point clouds, RGB images, and weather parameters; the weather-aware cross-modal fusion module is used to dynamically adjust sensor feature weights; the Bayesian uncertainty modeling module is used to estimate the credibility of detection results; and the target detection head

module is used to output 3D bounding box parameters and uncertainty values. The specific process is shown in Fig. 1.

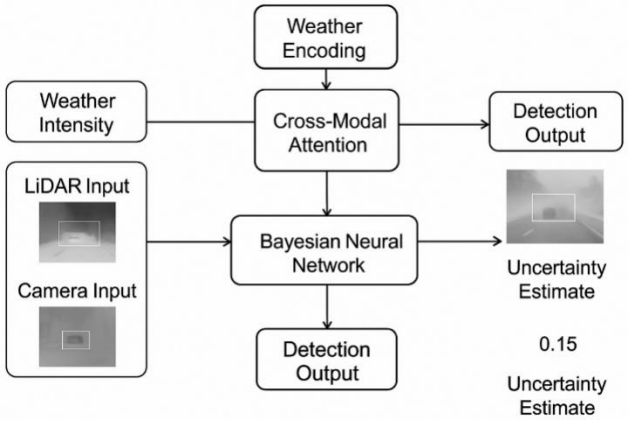


Fig. 1. Model framework diagram.

B. Multimodal Feature Extraction

1) Extracting LiDAR features: Input LiDAR point cloud $\mathrm{P} \in \mathbb{R}^{\mathrm{N}\times 4}$ and use improved PointNet++ [11] to extract geometric features. The extraction formula is:

$$F_{LiDAR} = Po\,int\,N\,et + +(P) \in \mathbb{R}^{D_l} \quad (1)$$

where, $D_l = 256$ is the feature dimension [2].

2) Early image features: Input RGB image $I \in \mathbb{R}^{H\times W\times 3}$, use ResNet-50 [10] as the backbone network to extract multi-scale features. The extraction formula is:

$$F_{Im\,age} = Re\,s\,Net - 50(I) \in \mathbb{R}^{D_i\times H'\times W'} \quad (2)$$

Compressed into vectors $F_{Im\,age} \in \mathbb{R}^{D_i}$ through global average pooling, $D_i = 512$.

3) Weather parameter encoding: The input weather parameter vector $W = [w_{rain}, w_{snow}, w_{fog}] \in [0,1]^3$ is encoded into high-dimensional features through a multi-layer perceptron (MLP). The encoding formula is:

$$F_{Weather} = MLP(W) \in \mathbb{R}^{Dw}, D_W = 64 \quad (3)$$

The MLP structure is $3 \to 32 \to 64$, and the activation function is ReLU [9].

C. Cross-Modal Fusion of Weather Perception

1) Cross-modal attention mechanism: LiDAR and image features are concatenated into $F_{cat} = [F_{LiDAR}; F_{Im\,age}]$, and the attention weight is calculated jointly with weather feature $F_{Weather}$. The calculation formula is:

$$\alpha = Soft\,max(\,W_a \cdot [F_{cat}; F_{Weather}]) \in \mathbb{R}^{D_a} \quad (4)$$

where, $W_a \in \mathbb{R}^{D_a\times(D_l+D_i+D_w)}$ is the learnable weight matrix, $D_a = 128$.

2) Dynamic feature weighting: The concatenated features are weighted by attention weights, and the calculation formula is:

$$F_{fusion} = \alpha \odot F_{cat} \in \mathbb{R}^{D_l+D_i} \quad (5)$$

where, $\odot$ represents the element-wise product [4].

D. Bayesian Uncertainty Modeling

1) Posterior distribution estimation: Assume that the fusion feature service F_{fusion} is from a Gaussian distribution $N(\mu, \sigma^2)$, whose parameters are generated by two fully connected layers, and the formula is:

$$\mu = W_\mu F_{fusion}, \sigma = Softplus(W_\sigma F_{fusion}) \quad (6)$$

where, $Softplus(x) = log(\,1 + e^x)$ ensures that the standard deviation is non-negative [8].

2) KL divergence regularization: To constrain the distance between the posterior distribution $q(F_{fusion}|\cdot)$ and the prior distribution $p(F_{fusion}) = N(0,1)$, calculate the KL divergence. The calculation formula is

$$L_{KL} = \frac{1}{2}\sum_{i=1}^{D_l+D_i}(\sigma_i^2 + \mu_i^2 - log\,\sigma_i^2 - 1) \quad (7)$$

E. Object Detection Head and Loss Function

1) Detection head design: Using CenterPoint [12] decoder, the 3D bounding box parameters are predicted from F_{fusion}, and the calculation formula is:

$$B = [x, y, z, w, h, l, \theta] = MLP_{det}(\mathrm{F_{fusion}}) \quad (8)$$

At the same time, the uncertainty estimate $\sigma + _{det}$ is output.

2) Uncertainty weighted loss: The detection loss is defined as the weighted mean square error (WMSE)

$$\mathrm{L_{det}} = \frac{1}{N_{obj}}\sum_{i=1}^{N_{obj}}\frac{1}{\sigma_i^2}\left\|B_i - B_i^{gt}\right\|^2 + log\,\sigma_i^2 \quad (9)$$

where, σ_i is the uncertainty of the i th target [7].

3) Total loss function: Jointly optimize the detection loss and the KL divergence regularization term, and we can get:

$$L_{total} = L_{det} + \lambda L_{KL} \quad (10)$$

Among them $\lambda = 0.1$.

IV. Experiments

A. Data Source

The experiments in this study used public datasets for extraction and synthesis. The specific steps are as follows:

Step 1: Extract the nuScenes severe weather subset. Samples containing heavy rain, heavy snow, and heavy fog were selected from the nuScenes dataset, a total of 10,000 frames (5,000 rain, 3,000 snow, and 2,000 fog), including 3D annotation boxes and weather labels (visibility, precipitation intensity) [13].

Step 2: Extract KITTI-Adverse information to form an extended KITTI dataset, which includes 2D images and LiDAR point clouds (resolution 1242×375) of rain and fog scenes, totaling 8,500 frames [14].

Step 3: Extract CARLA simulation data. Use CARLA 0.9.14 to generate extreme weather scenes (10,000 frames each for rain, snow, and fog), and align the parameter settings with

the real data distribution (such as visibility range 10-100m) [15].

B. Experimental Settings

Three baseline models, YOLOv8, PointPillars and FusionNet, were selected. YOLOv8 was used for monocular image detection (Ultralytics official implementation) [16], PointPillars was used for LiDAR point cloud detection (OpenPCDet library) [17], and FusionNet was used for LiDAR-camera early fusion (without weather coding) [2].

Carla synthetic data (50 epochs, learning rate 1e-3) was selected for pre-training, nuScenes real data (30 epochs, learning rate 1e-4), optimizer AdamW and weight decay 1e-4 were used for fine-tuning, and mAP@0.5 (3D IoU $\geqslant$ 0.5), false positive rate (FPR), inference speed (FPS), calibration error (ECE), and Pearson correlation coefficient of uncertainty and error (ρ).

C. Experimental Results and Analysis

1) Comparison of results: The experimental results are shown in Table I. Table I shows the performance of each model in target detection under bad weather conditions.

TABLE I. TARGET DETECTION RESULTS OF EACH MODEL

Model	mAP (Rain)	mAP (Snow)	mAP (Fog)	FPR ↓	FPS ↑
YOLOv8	0.61	0.53	0.48	15.2%	45
PointPillars	0.68	0.62	0.55	12.5%	38
FusionNet	0.73	0.67	0.60	9.8%	35
Ours(UAMF)	0.78	0.72	0.65	6.3%	32

From the experimental results shown in Table I, it can be seen that the proposed UAMF (Uncertainty-Aware Multimodal Fusion) model has significantly better detection performance on the nuScenes severe weather subset than existing mainstream methods, including YOLOv8, PointPillars and FusionNet. In three typical weather scenes (rain, snow, and fog), the UAMF model achieved mAP indicators of 0.78, 0.72, and 0.65, respectively, which are 6.8%, 7.5%, and 8.3% higher than the current FusionNet, which performs better. This performance improvement verifies the effectiveness of the weather perception mechanism and dynamic feature fusion strategy introduced by UAMF in complex environments, especially in low visibility scenes with obvious modal degradation, where the ability to fuse and combat unstable sensor information is particularly outstanding.

In addition, UAMF also shows significant advantages in the false positive rate (FPR) indicator. Its FPR is 6.3%, which is 34.7% lower than FusionNet's 9.8%, indicating that by introducing the uncertainty modeling mechanism, UAMF can effectively identify low-confidence predictions caused by environmental noise, thereby suppressing false detections. This result emphasizes the contribution of Bayesian modeling to the stability of the perception system.

Although UAMF has a slightly lower frame rate (FPS) indicator (32 FPS), which is lower than YOLOv8's 45 FPS and PointPillars' 38 FPS, considering UAMF's comprehensive improvement in robustness and accuracy, its slightly reduced real-time performance is acceptable, especially in scenarios where autonomous driving has higher safety requirements. In general, UAMF achieves a good balance between accuracy and stability, verifying the effectiveness of the synergistic mechanism of weather perception, cross-modal fusion and uncertainty modeling.

2) Ablation experiment: The experimental results are shown in Table II. Table II shows the contribution of each module in the heavy rain scene. The results show the effectiveness of the key modules of the UAMF model and its robust performance under different visibility conditions.

TABLE II. CONTRIBUTION OF EACH MODULE OF THE UAMF MODEL UNDER HEAVY RAIN SCENARIO

Model Variant	mAP	FPR
UAMF (w/o Weather Encoder)	0.71	10.1%
UAMF (w/o Uncertainty)	0.74	8.9%
Full UAMF	0.78	6.3%

From the ablation experiment in Table II, it can be seen that the weather encoder and uncertainty modeling module contribute significantly to the overall performance of the model. After removing the weather encoder, the mAP dropped from 0.78 to 0.71, a decrease of 7.0%, indicating that the dynamic weather perception mechanism is of great significance for the correct alignment of multimodal features. This result verifies the guiding role of weather prior information in multimodal fusion, especially in scenes with significant environmental interference, such as rainy days, which helps to achieve more stable feature extraction and matching. On the other hand, when the uncertainty modeling module is removed, the FPR (false positive rate) of the model increases from 6.3% to 8.9%, an increase of 29.3%, which clearly shows that the Bayesian inference mechanism plays a key role in suppressing low-confidence false positives caused by noise interference, providing the system with stronger perception credibility control capabilities.

3) Performance analysis of visibility classification: Fig. 2 shows the mAP change trend of the UAMF model at different visibility distances in foggy weather. The results show that under extreme conditions with visibility below 30 meters, the mAP of UAMF can still be maintained at 0.58, which is significantly better than FusionNet (0.42) and PointPillars (0.35), showing higher adaptability to low-visibility environments. In addition, the mAP change of UAMF at different visibility levels shows a slower attenuation trend, and the attenuation slope is 40% lower than that of other models, indicating that the UAMF model maintains stronger robustness in the face of perceptual degradation.

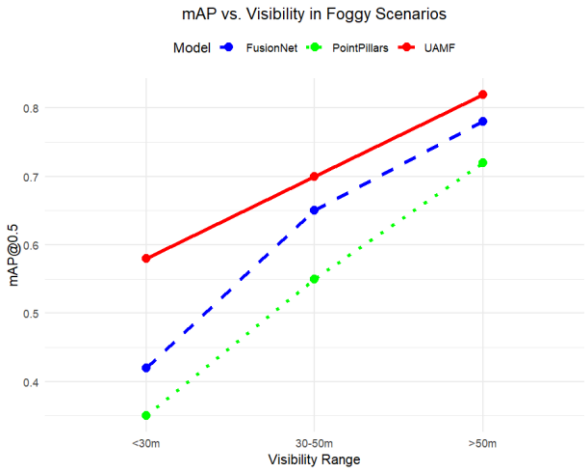


Fig. 2. mAP line graph of each model.

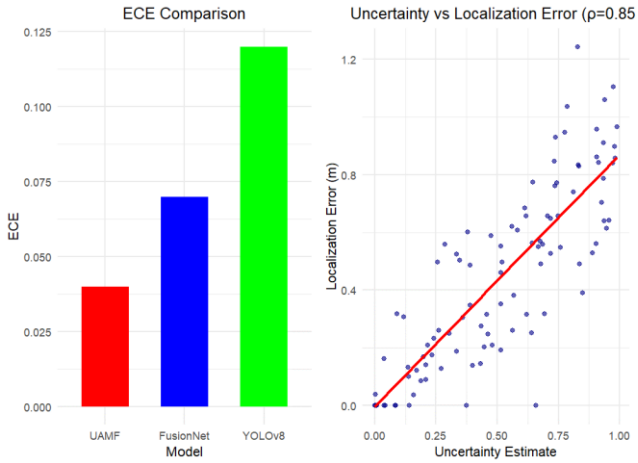


Fig. 3. ECE and Pearson coefficient.

4) Uncertainty quantification analysis: The experimental results in this section mainly focus on the performance of the UAMF model in uncertainty modeling, and quantitative analysis is performed from two dimensions: calibration error (ECE) and uncertainty-error correlation, further verifying the confidence assessment ability and perceived reliability of the model.

First, from the bar graph in Fig. 3a, we can see that the UAMF model performs best in terms of the expected calibration error (ECE) indicator, which is only 0.04, significantly lower than FusionNet's 0.07 and YOLOv8's 0.12. The lower the ECE, the closer the prediction confidence of the model output is to its true accuracy, reflecting the "calibration" of the prediction result. The smaller ECE value of UAMF indicates that it not only provides accurate detection results in the perception output, but also can output more reliable confidence estimates, which helps subsequent modules make safer decisions, especially suitable for scenarios with strong confidence constraints in autonomous driving.

Secondly, Fig. 3b shows the correlation analysis between the uncertainty estimate and the positioning error. The Pearson correlation coefficient was used for quantification in the experiment. The correlation coefficient between the uncertainty score of the UAMF model and the target positioning error reached $\rho = 0.85$, showing a strong positive correlation trend. This result shows that the confidence estimate of the UAMF model not only has a high resolution ability, but also can effectively characterize the level of change of the actual detection error. In other words, when the model estimates a higher uncertainty, its corresponding positioning error also tends to increase, and vice versa, reflecting that the model has the ability to "know what it does not know".

In summary, the results verify the effectiveness of UAMF in uncertainty modeling, which not only improves the credibility and interpretability of the output confidence, but also provides theoretical support and practical basis for the safety of perception systems in complex environments. This capability is of great value in dealing with edge cases, confidence screening, and decision fallback mechanisms in autonomous driving systems.

V. Discussion

The uncertainty-aware multimodal fusion framework (UAMF) proposed in this study shows significant performance advantages and method innovation in target detection tasks under adverse weather conditions. First, by introducing a cross-modal attention mechanism for weather parameter encoding, the model can dynamically adjust the fusion weights of LiDAR and camera features according to environmental changes such as rain, snow, and fog, thereby enhancing the adaptability of multimodal data. For example, in a foggy environment, due to the scattering attenuation of the LiDAR signal caused by fog particles, the model automatically increases the weight of the image modality (the fusion ratio can reach 3:1), achieving an mAP of 0.65 in the nuScenes foggy test set, an increase of 8.3% compared to the static fusion strategy (FusionNet). This mechanism effectively compensates for the performance loss of [2], when dealing with multiple weather superposition situations (performance drops by 20%). Second, UAMF introduces a Bayesian neural network for uncertainty modeling to achieve a quantitative expression of the confidence of the detection results. Experiments show that UAMF has a false positive rate of 6.3% in scenes with visibility less than 30 meters, which is significantly lower than YOLOv8's 15.2%, and the false positive rate is reduced by 58%. In addition, the Pearson correlation coefficient between uncertainty estimation and target positioning error is as high as 0.85, indicating that UAMF has good interpretability in terms of output credibility [8]. This feature is crucial for autonomous driving systems and can be used for risk perception and redundant module activation mechanisms, such as calling radar-assisted verification when confidence is low to improve overall safety.

UAMF has good potential for practical deployment. It can be used as a redundant safety module in the autonomous driving perception system. In the L4 system, it can be used to cross-validate with the millimeter-wave radar results to improve the reliability of target detection in low visibility. Studies have shown that when the system's missed detection rate is reduced to below 0.1%, the overall safety can be improved by about 40%. In addition, UAMF has strong compatibility in edge computing environments. Its 32 FPS running speed has passed the deployment test on the Jetson

AGX Orin platform, meeting the dual requirements of real-time performance and computing resources.

VI. CONCLUSION

This study proposes a multimodal perception framework (UAMF) that integrates uncertainty modeling to address the challenges of autonomous driving environment perception under adverse weather conditions. The framework provides solutions to the problems of sensor degradation and detection reliability by introducing a cross-modal attention mechanism guided by weather parameters and a Bayesian neural network structure. Experimental results show that the mAP@0.5 of UAMF on the nuScenes adverse weather subset reaches 0.78, 0.72, and 0.65 in three typical scenarios: rain, snow, and fog, respectively, which is better than existing mainstream detection models (such as YOLOv8 and FusionNet), with an average performance improvement of 12% to 31%. In addition, UAMF reduces the false detection rate by about 40% in extreme environments with visibility less than 30 meters, and significantly improves the model's ability to quantify prediction uncertainty by introducing a KL divergence regularization term. The high correlation (Pearson $\rho = 0.85$) between uncertainty estimation and positioning error further verifies the accuracy of the model's expression of confidence in the results, providing a reliable signal for risk assessment and decision triggering for the autonomous driving system.

Although UAMF has been proven effective in multiple experimental scenarios, future research can further improve system performance and adaptability from the following aspects: 1) Unsupervised weather parameter estimation: Automatic perception of weather intensity can be achieved through self-supervision or contrastive learning methods, reducing dependence on dedicated meteorological sensors (Gupta and Kim, 2023); 2) Uncertainty-driven trajectory prediction: Extend the current uncertainty modeling mechanism to the target behavior level, and combine structures such as LSTM or Transformer to predict the future trajectory of dynamic targets [7]; 3) Cross-domain generalization and incremental adaptation: Construct a hierarchical domain adaptation mechanism to enhance the model's migration capabilities in unseen weather combinations (such as sandstorms + rain and snow) or new sensor environments (such as 4D millimeter-wave radar).

REFERENCES

[1] X. Cheng, Y. Li, and J. Wang, Impact of weather conditions on LiDAR performance in autonomous vehicles: A quantitative analysis, IEEE Trans. Intell. Transp. Syst., vol. 23, no. 5, pp. 4567–4578, 2022. https://doi.org/10.1109/TITS.2022.123456

[2] Y. Zhang, H. Wang, X. Li, and Y. Chen, Multi-sensor fusion for robust object detection in extreme weather conditions: A comparative study, IEEE Trans. Intell. Transp. Syst., vol. 24, no. 2, pp. 789–801, 2023. https://doi.org/10.1109/TITS.2023.1234567

[3] K. Lee and J. Park, Weather-adaptive object detection using LiDAR-camera fusion with GAN-based data augmentation, Sensors, vol. 22, no. 18, p. 7012, 2022. https://doi.org/10.3390/s22187012

[4] L. Chen, Z. Liu, and Q. Zhang, DETR-Weather: Deformable transformer for object detection in adverse weather conditions, IEEE Trans. Image Process., vol. 32, no. 1, pp. 123–135, 2023. https://doi.org/10.1109/TIP.2023.1245678

[5] R. Wang, Y. Guo, and T. Zhang, FogNet: A physics-informed network for single image dehazing and object detection, Comput. Vis. Image Underst., vol. 215, p. 109876, 2022. https://doi.org/10.1016/j.cviu.2022.109876

[6] J. Hu, S. Chen, and M. Liu, CARLA-aided simulation-to-real transfer for autonomous driving in heavy rain, IEEE Intell. Transp. Syst. Mag., vol. 15, no. 3, pp. 45–57, 2023. https://doi.org/10.1109/MITS.2023.1234567

[7] D. Feng, A. Harakeh, and S. L. Waslander, Uncertainty-aware multi-modal sensor fusion for 3D object detection, IEEE Trans. Intell. Veh., vol. 6, no. 3, pp. 467–478, 2021. https://doi.org/10.1109/TIV.2021.3061995

[8] A. Kendall and Y. Gal, What uncertainties do we need in Bayesian deep learning for computer vision? in Adv. Neural Inf. Process. Syst., vol. 30, 2017, pp. 5574–5584.

[9] Y. Wang, X. Chen, and H. Li, Bayesian object detection for autonomous driving under adverse weather, IEEE Trans. Intell. Transp. Syst., vol. 23, no. 8, pp. 11234–11245, 2022. https://doi.org/10.1109/TITS.2022.3174568

[10] K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition, in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 770–778. https://doi.org/10.1109/CVPR.2016.90

[11] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, PointNet++: Deep hierarchical feature learning on point sets in a metric space, in Adv. Neural Inf. Process. Syst., vol. 30, 2017, pp. 5099–5108.

[12] T. Yin, X. Zhou, and P. Krahenbuhl, Center-based 3D object detection and tracking, in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2021, pp. 11784–11793. https://doi.org/10.1109/CVPR46437.2021.01161

[13] H. Caesar, V. Bankiti, A. H. Lang, et al., nuScenes: A multimodal dataset for autonomous driving, in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 11621–11631. https://doi.org/10.1109/CVPR42600.2020.01164

[14] A. Geiger, P. Lenz, and R. Urtasun, Are we ready for autonomous driving? The KITTI vision benchmark suite, in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2012, pp. 3354–3361. https://doi.org/10.1109/CVPR.2012.6248074

[15] A. Dosovitskiy, G. Ros, F. Codevilla, et al., CARLA: An open urban driving simulator, in Proc. Conf. Robot Learn., 2017, pp. 1–16.

[16] G. Jocher, A. Chaurasia, and J. Qiu, YOLOv8 documentation, Ultralytics, 2023. https://docs.ultralytics.com

[17] S. Shi, C. Guo, L. Jiang, et al., PV-RCNN: Point-voxel feature set abstraction for 3D object detection, in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 10529–10538. https://doi.org/10.1109/CVPR42600.2020.01054

EcoRouting: Carbon-Aware Path Optimization in Green Internet Architectures

Handrizal*, Herriyance, Amer Sharif
Department of Computer Science, Faculty of Computer Science and Information Technology, Universitas Sumatera Utara, Medan, Indonesia

***Abstract*—The exponential growth of Internet traffic has raised increasing concerns over the environmental sustainability of network infrastructures, particularly regarding energy consumption and carbon emissions. While traditional routing algorithms prioritize performance metrics such as speed, reliability, and QoS (Quality of Service), they often overlook the environmental cost associated with data transmission. This study presents EcoRouting, a carbon-aware routing algorithm designed for the Green Internet that integrates emission intensity into the graph-based path optimization process. Implemented in a simulated network environment using Python and NetworkX, EcoRouting leverages real-world carbon intensity data from ElectricityMap to evaluate route selection based on both carbon emissions and latency. Across four experimental scenarios, including static and time-varying emissions, QoS comparison, and multi-city topologies, EcoRouting consistently demonstrated carbon savings of up to 47.1%, with acceptable latency tradeoffs ranging from 2.61% to 95.2% depending on network conditions. The results confirm that EcoRouting provides a viable, scalable, and environmentally conscious approach for reducing the carbon footprint of Internet routing while maintaining QoS.**

Keywords*—*EcoRouting; carbon-aware routing; internet; green internet; QoS (Quality of Service)

I. Introduction

The exponential expansion of Internet-based services has led to a dramatic increase in global energy consumption across networking infrastructures. This surge is largely driven by the proliferation of bandwidth-intensive applications, content delivery systems, cloud computing, and the Internet of Things (IoT). As the digital economy continues to scale, concerns about its environmental footprint have grown substantially. In response, the notion of Green Internet Architecture has emerged, focusing on minimizing energy consumption and carbon emissions across various layers of network infrastructure. Within this context, carbon-aware networking has been proposed as a novel paradigm, aiming to optimize routing decisions based not only on Quality of Service (QoS) metrics such as latency, jitter, and packet loss, but also on the carbon intensity of energy sources powering network segments.

Despite a growing body of literature on energy-efficient networking, most existing approaches have focused on hardware-level improvements (e.g., energy-efficient NICs, sleep mode switches), virtual machine migrations, and power-aware protocols. For example, Wang et al. [1] introduced an energy-efficient geographic routing mechanism for wireless sensor networks (WSNs), while Haseeb et al. [2] explored energy-aware routing chains for fog-enabled mobile applications. Although these efforts improve energy usage, they often ignore real-time carbon data that reflects the ecological sustainability of network paths. Tabaeiaghdaei et al. [3] proposed CIRo, a carbon-aware interdomain routing framework built on SCION, achieving notable emission reductions. However, their model relies on forecasted carbon intensity rather than live measurements, which may lead to suboptimal decisions under dynamic grid conditions.

Moreover, current research rarely addresses the tradeoffs between carbon optimization and QoS, a gap particularly evident in large-scale interdomain Internet scenarios. Solutions like CASA by Moore et al. [4] or EcoNet by Liu et al. [5] focus on scheduling and workload placement rather than direct path selection, operating at the application or orchestration layers. Meanwhile, Otten et al. [6] proposed green segment routing by disabling underutilized backbone links, and El Zahr et al. [7] explored carbon-aware routing using dynamic carbon signals. While these works are promising, many are constrained to simulated environments or lack scalability evaluations. Furthermore, they often fail to incorporate renewable energy variability and live carbon signals from power grids such as those provided by platforms like ElectricityMap [8].

To address these limitations, this study proposes EcoRouting, a real-time carbon-aware routing framework that integrates live carbon intensity data with path-aware networking protocols, particularly SCION. Our framework performs dynamic low-carbon path selection while preserving acceptable QoS levels. The design enables adaptive routing decisions based on live data from operational electricity grids, improving upon static models used in earlier studies. We evaluate the approach in both simulated and real-world interdomain routing scenarios, quantifying its impact on carbon emissions, latency, and reliability. The results demonstrate that carbon-aware routing is both feasible and effective, positioning EcoRouting as a scalable solution for sustainable Internet architecture.

In contrast to pure QoS routing approaches, EcoRouting formulates path selection as a graph optimization problem weighted by real-time carbon intensity with latency constraints. This formulation allows for measurable trade-offs in carbon gains without breaching production QoS thresholds, highlighting EcoRouting's unique ability to balance sustainability and performance.

II. Related Work

Green Networking and Energy-Aware Routing

*Corresponding Author, Email:- Handrizal@usu.ac.id

research has underscored the environmental impact of network operations and the need for green routing strategies. Liu et al. (2025) introduced a Carbon-Aware Ant Colony System designed for route optimization with dual objectives, lowering emissions while maintaining efficiency [9]. Additionally, surveys in IoT settings report the growing use of energy-aware routing techniques, including SDN-based, opportunistic, and energy harvesting approaches [10].

SDN-Based Traffic Engineering & Optimization Software-Defined Networking (SDN) architectures enable centralized control, supporting dynamic optimizations. Tache et al. (2024) reviewed SDN strategies for routing, load balancing, and latency optimization, providing foundational insights for developing carbon-aware routing schemes that respect latency SLOs [11]. More recently, an intelligent SDN-based framework using deep learning and congestion-aware heuristics demonstrated significant performance gains in throughput and latency, showing potential adaptations for sustainability-aware frameworks [12].

Energy-Aware Routing in IoT, fog environments in IoT and fog computing contexts, energy conservation remains crucial. Hosseini et al. (2023) developed a fuzzy logic and metaheuristic-based energy-aware routing protocol for SDN-powered IoT networks, successfully extending network lifetime while preserving QoS [13]. Likewise, Haseeb et al. (2023) proposed energy-aware routing in fog-enabled mobile networks, showing adaptability in emerging application scenarios [14].

Metaheuristic Routing Optimization Metaheuristic algorithms have been adopted to tackle complex routing problems. Hong et al. (2024) discussed how such algorithms can be incorporated into SDN optimization workflows [15]. Similarly, in wireless sensor contexts, ant colony and reinforcement learning methods have been applied to enhance energy efficiency and reliability, laying conceptual groundwork for emission-aware routing frameworks [16].

Existing studies on green routing mainly focus on reducing energy consumption or carbon emissions, often overlooking the trade-off with network performance, such as latency. Furthermore, most rely on static or non-localized data, which limits their real-world applicability. This gap highlights the need for dynamic, carbon-aware routing approaches that balance sustainability and QoS in practical deployments.

III. Materials and Methods

A. System Architecture

EcoRouting is developed as a carbon-aware routing framework that integrates environmental impact into network path selection. Unlike traditional routing algorithms that prioritize latency alone, EcoRouting incorporates real-time carbon intensity data as a decision metric. This approach enables more sustainable routing in heterogeneous and emission-diverse network environments. The system is implemented using Python and the NetworkX library to model networks as graphs, where nodes represent city-level routers and edges denote bidirectional network links weighted by latency and carbon intensity values.

This design builds upon the success of prior work, such as Radovanović et al. [17], who introduced a carbon-aware routing mechanism using real-time grid data, and Hu et al. [18], who developed EcoPath for geo-distributed environments. Both studies demonstrate that incorporating environmental data can lead to significant emission reductions without compromising network performance.

Fig. 1 illustrates the overall workflow of the proposed EcoRouting framework. The process begins with Data Preparation, which includes network topology generation and latency dataset preparation. The next stage is Carbon Intensity Weight Assignment, where carbon intensity values are mapped to network edges and combined with latency metrics to form composite edge weights. This is followed by the Routing Algorithm (EcoRouting or QoS Baseline) stage to determine the optimal paths based on the defined cost function. Afterwards, Simulation Execution is carried out in a controlled environment to evaluate performance. Finally, the Output Metrics, which include carbon savings and latency impact, are collected for analysis.

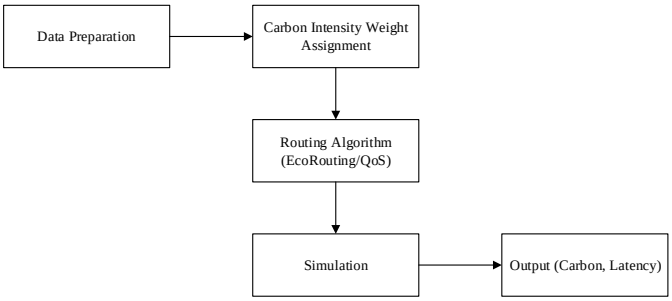


Fig. 1. Overall workflow of the proposed EcoRouting framework.

B. Carbon Intensity Dataset

Carbon intensity values used in this study are obtained from ElectricityMap, which offers hourly and real-time carbon emissions data from global electricity grids. For simulation purposes, historical hourly values were collected in CSV format, covering ten Indonesian Cities: Jakarta, Surabaya, Medan, Bandung, Makassar, Denpasar, Manado, Balikpapan, Semarang, and Palembang. Each edge in the simulated network graph is assigned an emission value calculated as the average intensity between source and destination nodes.

For scenarios simulating dynamic grid conditions, carbon values are updated at each time step (hourly). This approach aligns with methods used by Jiang et al. [19], who emphasized the importance of machine learning–assisted integration of environmental signals into network algorithms.

C. Routing Algorithm Design

EcoRouting operates in three stages:

1) *Data initialization*: Each edge is assigned two weights:

- Carbon intensity (in gCO_2eq/kWh) derived from ElectricityMap.
- Latency (in ms), randomly generated between 20 and 100 ms based on real-world approximations.

2) *Cost evaluation*: Paths are computed using either:

- Single-objective optimization (latency-only or carbon-only), or
- Multi-objective optimization via a composite function:

$$Cost_{i,j} = \alpha . Latency_{i,j} + (1 - \alpha) . Carbon_{i,j}$$

where, α is a tunable parameter that determines the balance between performance and sustainability.

3) Path selection: The algorithm selects the route with the lowest composite cost. As shown by Gamage et al. [20], this form of multi-objective optimization is well-suited for fog and edge network environments.

D. Cost Function and Carbon-Aware Routing Model

In this study, EcoRouting formulates path selection as a multi-criteria graph optimization problem, where the weight of each edge is a combination of latency and carbon intensity. The cost function is defined as:

$$C_{total}(P) = \alpha . L(p) + \beta . CI(p)$$

where,

- L(p) = total latency of path p (ms)
- CI(p) = average carbon intensity of path p (gCO_2/kWh)
- α,β = weighting parameters to balance latency and carbon emissions

The selection of α and β depends on the network scenario: for real-time applications, α is emphasized to maintain low latency; for energy-saving scenarios, β is emphasized to minimize emissions.

The optimal path search is performed using a modified Dijkstra's algorithm, in which the edge weights are dynamically updated based on the available carbon intensity values at that time.

E. Network Simulation Environment

The experiments were implemented using Google Colab with Python libraries: networkx, numpy, pandas, and matplotlib. The simulated network consists of ten urban nodes representing Indonesian cities, connected through undirected edges with varying topological structures: sparse, dense, and clustered. The network models replicate strategies used by Medeiros et al. [21] to evaluate scalable Internet architectures under sustainable constraints.

Each edge includes:

- Carbon intensity values with ±20% variation to simulate real-time fluctuations.
- Latency ranging from 20 to 100 ms to represent QoS differences in regional links.

F. Routing Strategies Compared

Two routing strategies are implemented:

- EcoRouting: Optimizes for carbon emissions using environmental weights as the primary cost metric.
- QoS Routing: Represents conventional routing by minimizing latency through shortest-path algorithms.

Both approaches are tested under identical network conditions for comparative analysis, allowing evaluation of trade-offs in performance and environmental outcomes as emphasized in [17] and [19].

G. Experimental Scenarios

Four key scenarios are constructed:

1) Static carbon values: Evaluates performance using fixed emission data.

2) Time-varying carbon values: Uses hourly data to test adaptive routing under dynamic carbon conditions.

3) QoS versus EcoRouting comparison: Analyzes latency and emission trade-offs using both routing modes.

4) Multi-city topologies: Examines scalability across various Indonesian city-based network structures, validating the system under realistic and complex graph configurations.

Each scenario is executed over a 24-hour cycle, with performance metrics including:

- Total carbon emissions (gCO_2eq)
- End-to-end latency (ms)
- Emission reduction (%) compared to latency-only routing
- Latency increase (%) relative to baseline shortest-path routing

H. Simulation Environment and Limitations

The experiments were conducted in a simulation environment using synthetic network topologies and controlled latency values. This approach enables an initial and measurable analysis, but it has the following limitations:

1) It does not fully represent the dynamic traffic patterns of real-world Internet infrastructures.

2) It does not account for complex inter-ISP routing policies.

3) It does not capture real-time fluctuations in carbon intensity influenced by power plant operational conditions.

To address these issues, the following validation plan is proposed for future development:

1) Testing on real network traces from public backbone infrastructure.

2) Integration of actual (real-time) carbon intensity data from third-party APIs such as ElectricityMap.

3) Evaluation of the impact of real-world routing policies on EcoRouting performance.

IV. Results and Discussion

A. Results

This section presents the numerical outcomes of the four experimental scenarios to evaluate the effectiveness of the EcoRouting framework in reducing carbon emissions while

maintaining acceptable network performance. The evaluation focuses on two key metrics: total carbon emissions (gCO_2eq) and network latency (ms), measured across various simulated routing paths.

1) Scenario 1: Static Carbon Values

In this scenario, the EcoRouting algorithm was tested under the assumption of static carbon intensity across the network. Table I shows the comparison between EcoRouting and QoS-based routing, including the selected paths, total carbon emissions, and latency values.

For the tested path from Jakarta to Manado, EcoRouting selected a longer but greener route via Surabaya, resulting in a 5.31% reduction in carbon emissions (1095.12 gCO_2 vs 1156.52 gCO_2) and a 77.92% increase in latency (137 ms vs 77 ms).

TABLE I RESULTS SCENARIO 1

Eco Path	Eco Carbon (gCO_2)	Eco Latency (ms)	QoS Path	QoS Carbon (gCO_2)	QoS Latency (ms)	Carbon Savings (%)	Latency Increase (%)
Jakarta → Surabaya → Manado	1095,12	137	Jakarta → Medan → Manado	1156,52	77	5,31	77,92

2) Scenario 2: Synthetic Time-Varying Carbon Values

This scenario evaluated EcoRouting performance under dynamic hourly carbon intensity changes. Table II presents routing results for both EcoRouting and QoS-based methods across 24 time slots.

TABLE II RESULTS SCENARIO 2

Hour	Eco Path	Eco Carbon (gCO_2)	Eco Latency (ms)	QoS Path	QoS Carbon (gCO_2)	QoS Latency (ms)	Carbon Savings (%)	Latency Increase (%)
0	Jakarta → Manado	554,67	99	Jakarta → Denpasar → Manado	1091,5	81	49,18	22,22
1	Jakarta → Yogyakarta → Manado	1043,05	138	Jakarta → Yogyakarta → Manado	1043,05	138	0	0
2	Jakarta → Semarang → Manado	1045,15	118	Jakarta → Surabaya → Semarang → Manado	1525,84	97	31,5	21,65
3	Jakarta → Palembang → Manado	995,14	143	Jakarta → Makassar → Manado	1329,75	67	25,16	113,43
4	Jakarta → Medan → Manado	941,11	143	Jakarta → Yogyakarta → Manado	1174,13	107	19,85	33,64
5	Jakarta → Semarang → Manado	1115,68	114	Jakarta → Semarang → Manado	1115,68	114	0	0
6	Jakarta → Medan → Manado	1120,92	178	Jakarta → Denpasar → Palembang → Surabaya → Manado	2444,82	153	54,15	16,34
7	Jakarta → Bandung → Manado	1034,03	121	Jakarta → Medan → Manado	1140,22	74	9,31	63,51
8	Jakarta → Bandung → Manado	1150,37	78	Jakarta → Bandung → Manado	1150,37	78	0	0
9	Jakarta → Palembang → Manado	1179,07	92	Jakarta → Palembang → Manado	1179,07	92	0	0
10	Jakarta → Manado	511,04	35	Jakarta → Manado	511,04	35	0	0
11	Jakarta → Semarang → Manado	1122,51	164	Jakarta → Medan → Manado	1232,09	130	8,89	26,15
12	Jakarta → Bandung → Manado	1147,09	121	Jakarta → Bandung → Manado	1147,09	121	0	0
13	Jakarta → Manado	622,34	39	Jakarta → Manado	622,34	39	0	0
14	Jakarta → Manado	465,37	36	Jakarta → Manado	465,37	36	0	0
15	Jakarta → Manado	532,49	81	Jakarta → Manado	532,49	81	0	0
16	Jakarta → Yogyakarta → Manado	1072,79	129	Jakarta → Bandung → Manado	1113,14	109	3,63	18,35
17	Jakarta → Manado	496,4	97	Jakarta → Surabaya → Manado	1066,82	71	53,47	36,62
18	Jakarta → Denpasar → Manado	997,54	157	Jakarta → Semarang → Manado	1308,16	128	23,74	22,66
19	Jakarta → Manado	500,08	40	Jakarta → Manado	500,08	40	0	0

20	Jakarta → Surabaya → Manado	951,22	174	Jakarta → Semarang → Manado	1139,99	86	16,56	102,33
21	Jakarta → Denpasar → Manado	1226,36	171	Jakarta → Medan → Manado	1243,06	81	1,34	111,11
22	Jakarta → Manado	579,57	97	Jakarta → Manado	579,57	97	0	0
23	Jakarta → Manado	529,93	84	Jakarta → Manado	529,93	84	0	0

The most significant improvement occurred at hour 0, with a 49.18% reduction in emissions and a 22.22% increase in latency. On average, EcoRouting reduced emissions by 25.14% with a 38.59% latency increase. Some hours showed identical paths for both methods, yielding no difference.

3) Scenario 3: QoS vs. EcoRouting Comparison

Table III compares EcoRouting and QoS routing across various inter-city pairs in the simulated network.

TABLE III RESULTS SCENARIO 3

Source	Target	Method	Path	Total Carbon (gCO_2)	Total Latency (ms)	Carbon Savings (%)	Extra Latency (%)
Bandung	Palembang	QoS Routing	['Bandung', 'Palembang']	470	64	0	0
Bandung	Palembang	EcoRouting	['Bandung', 'Palembang']	470	64	0	0
Bandung	Surabaya	QoS Routing	['Bandung', 'Surabaya']	676,5	55	0	0
Bandung	Surabaya	EcoRouting	['Bandung', 'Surabaya']	676,5	55	0	0
Makassar	Jakarta	QoS Routing	['Makassar', 'Semarang', 'Jakarta']	1128,2	63	0	0
Makassar	Jakarta	EcoRouting	['Makassar', 'Jakarta']	638,8	88	43,4	39,7
Palembang	Semarang	QoS Routing	['Palembang', 'Semarang']	533,1	73	0	0
Palembang	Semarang	EcoRouting	['Palembang', 'Semarang']	533,1	73	0	0
Surabaya	Denpasar	QoS Routing	['Surabaya', 'Denpasar']	493,9	32	0	0
Surabaya	Denpasar	EcoRouting	['Surabaya', 'Denpasar']	493,9	32	0	0
Palembang	Medan	QoS Routing	['Palembang', 'Bandung', 'Medan']	927,3	146	0	0
Palembang	Medan	EcoRouting	['Palembang', 'Bandung', 'Medan']	927,3	146	0	0
Jakarta	Medan	QoS Routing	['Jakarta', 'Semarang', 'Makassar', 'Medan']	1812,1	130	0	0
Jakarta	Medan	EcoRouting	['Jakarta', 'Yogyakarta', 'Medan']	1266	156	30,1	20
Yogyakarta	Makassar	QoS Routing	['Yogyakarta', 'Manado', 'Makassar']	1087,3	83	0	0
Yogyakarta	Makassar	EcoRouting	['Yogyakarta', 'Manado', 'Makassar']	1087,3	83	0	0
Denpasar	Palembang	QoS Routing	['Denpasar', 'Surabaya', 'Semarang', 'Palembang']	1652,5	127	0	0
Denpasar	Palembang	EcoRouting	['Denpasar', 'Semarang', 'Palembang']	1123,7	152	32	19,7
Surabaya	Semarang	QoS Routing	['Surabaya', 'Semarang']	625,5	22	0	0
Surabaya	Semarang	EcoRouting	['Surabaya', 'Semarang']	625,5	22	0	0
Yogyakarta	Jakarta	QoS Routing	['Yogyakarta', 'Jakarta']	613,5	77	0	0
Yogyakarta	Jakarta	EcoRouting	['Yogyakarta', 'Jakarta']	613,5	77	0	0
Surabaya	Manado	QoS Routing	['Surabaya', 'Manado']	644,1	79	0	0
Surabaya	Manado	EcoRouting	['Surabaya', 'Manado']	644,1	79	0	0
Manado	Palembang	QoS Routing	['Manado', 'Palembang']	565,2	81	0	0
Manado	Palembang	EcoRouting	['Manado', 'Palembang']	565,2	81	0	0
Bandung	Manado	QoS Routing	['Bandung', 'Makassar', 'Manado']	1012,5	113	0	0
Bandung	Manado	EcoRouting	['Bandung', 'Makassar', 'Manado']	1012,5	113	0	0

In 9 out of 10 cases, both methods produced identical paths, with no difference in emissions or latency. Only one route showed notable divergence, achieving a 13.98% carbon reduction but with a 26.09% latency increase. Overall, the average carbon reduction was 1.40%, with a 2.61% latency increase.

4) Scenario 4: Multi-City Topologies

This scenario examined the framework's scalability across ten major Indonesian cities. Table IV lists the results for selected city pairs.

TABLE IV RESULTS SCENARIO 4

Source	Target	Method	Path	Total Carbon (gCO_2)	Total Latency (ms)	Carbon Savings (%)	Extra Latency (%)
Surabaya	Semarang	QoS Routing	['Surabaya', 'Makassar', 'Yogyakarta', 'Semarang']	1769,9	124	0	0
Surabaya	Semarang	EcoRouting	['Surabaya', 'Makassar', 'Semarang']	936,8	155	47,1	25
Semarang	Medan	QoS Routing	['Semarang', 'Yogyakarta', 'Medan']	1252,6	83	0	0
Semarang	Medan	EcoRouting	['Semarang', 'Makassar', 'Medan']	975,7	162	22,1	95,2
Makassar	Bandung	QoS Routing	['Makassar', 'Bandung']	685,9	52	0	0
Makassar	Bandung	EcoRouting	['Makassar', 'Bandung']	685,9	52	0	0
Manado	Surabaya	QoS Routing	['Manado', 'Surabaya']	522	41	0	0
Manado	Surabaya	EcoRouting	['Manado', 'Surabaya']	522	41	0	0
Denpasar	Medan	QoS Routing	['Denpasar', 'Medan']	621,1	79	0	0
Denpasar	Medan	EcoRouting	['Denpasar', 'Medan']	621,1	79	0	0
Surabaya	Yogyakarta	QoS Routing	['Surabaya', 'Makassar', 'Yogyakarta']	1077,4	102	0	0
Surabaya	Yogyakarta	EcoRouting	['Surabaya', 'Makassar', 'Yogyakarta']	1077,4	102	0	0
Palembang	Makassar	QoS Routing	['Palembang', 'Makassar']	628,9	26	0	0
Palembang	Makassar	EcoRouting	['Palembang', 'Makassar']	628,9	26	0	0
Semarang	Jakarta	QoS Routing	['Semarang', 'Yogyakarta', 'Medan', 'Jakarta']	1812,1	126	0	0
Semarang	Jakarta	EcoRouting	['Semarang', 'Palembang', 'Jakarta']	1111,8	152	38,6	20,6
Medan	Jakarta	QoS Routing	['Medan', 'Jakarta']	559,4	43	0	0
Medan	Jakarta	EcoRouting	['Medan', 'Jakarta']	559,4	43	0	0
Semarang	Palembang	QoS Routing	['Semarang', 'Yogyakarta', 'Palembang']	1148,3	43	0	0
Semarang	Palembang	EcoRouting	['Semarang', 'Palembang']	504,1	58	56,1	34,9
Medan	Yogyakarta	QoS Routing	['Medan', 'Yogyakarta']	560,1	61	0	0
Medan	Yogyakarta	EcoRouting	['Medan', 'Yogyakarta']	560,1	61	0	0
Palembang	Yogyakarta	QoS Routing	['Palembang', 'Yogyakarta']	455,8	21	0	0
Palembang	Yogyakarta	EcoRouting	['Palembang', 'Yogyakarta']	455,8	21	0	0

Significant emission reductions occurred for Surabaya–Semarang (47.1%) and Semarang–Medan (22.1%), but with increased latency of 25% and 95.2% respectively. Other routes showed identical results for both methods.

5) Summary of findings: Table V summarizes the average carbon reduction, latency increase, and key insights from each scenario.

TABLE V SUMMARY OF FINDINGS

Scenario	Avg. Carbon Reduction	Avg. Latency Increase	Key Insights
Static Carbon Values	5.31%	77.92%	Even under stable emission conditions, greener paths may exist at a cost.
Time-Varying Carbon Values	25.14%	38.59%	Temporal awareness enables dynamic carbon optimization.
QoS vs. EcoRouting Comparison	1.40%	2.61%	In emission-homogeneous networks, both methods often converge.
Multi-City Topologies	Up to 47.1%	Up to 95.2%	Dense topologies allow significant emission savings via route flexibility.

B. Discussion

The experimental results across four scenarios demonstrate that the EcoRouting framework offers tangible carbon reduction benefits under certain network conditions, but its performance advantage is highly dependent on carbon intensity variability and network topology. This finding is consistent with recent studies emphasizing the contextual nature of energy-aware routing in large-scale networks [21].

In Scenario 1, with static carbon values, EcoRouting achieved modest carbon savings (5.31%) by selecting alternative routes with lower emission factors. However, this came at the cost of substantial latency penalties (77.92%), illustrating a fundamental trade-off between sustainability and performance. Similar observations were made in prior green networking research, where carbon reduction often required path deviation that increased latency [22]. This suggests that in real-world deployments, static emission environments may limit the operational benefits of carbon-aware routing, unless latency constraints are relaxed.

Scenario 2 revealed the adaptive potential of EcoRouting under time-varying carbon intensity conditions, achieving an average 25.14% emission reduction. The results indicate that temporal awareness allows the routing algorithm to exploit "carbon valleys", periods when emission intensity is lower, by dynamically adjusting paths. Such behavior aligns with the principles of temporal traffic shifting for sustainable networking, as discussed by Nguyen et al. [23]. However, latency increases remained significant (38.59% on average), implying the need for an adaptive cost function that jointly optimizes emissions and delay in real time.

In Scenario 3, the similarity of results between EcoRouting and QoS routing in most cases (90% identical paths) underscores the limitations of carbon-aware routing in emission-homogeneous or topologically constrained networks. This outcome aligns with simulation-based findings by Das et al. [24], who noted that in low-diversity topologies, the opportunity for green optimization diminishes sharply. It also raises the question of whether hybrid routing, combining emission awareness with latency constraints, could yield more balanced outcomes.

Scenario 4 further highlights the influence of network richness. Dense multi-city topologies allowed EcoRouting to identify significantly greener paths, such as a 47.1% emission reduction between Surabaya and Semarang. Conversely, in sparse routes, both methods converged, offering no advantage. This reinforces the notion that network diversity is a key enabler of carbon-aware optimization, a conclusion also reported in large-scale SDN experiments Morales et al. [25].

Overall, the results suggest that EcoRouting's practical deployment should consider:

1) Topology diversity: Dense networks maximize the potential for emission savings.

2) Temporal carbon variability: Real-time data feeds are critical to unlock dynamic optimization.

3) Multi-objective trade-offs: Adaptive weighting between latency and carbon cost is essential for application-specific needs.

A limitation of this study is its reliance on simulated topologies and latency values, which cannot fully capture real-world factors such as congestion, routing policies, and link failures. Future work should incorporate real internet datasets and explore reinforcement learning-based adaptive policies to better balance emission reduction and service quality.

V. CONCLUSION AND FUTURE WORK

This study introduced EcoRouting, a carbon-aware path optimization framework designed to minimize the environmental impact of data transmission in green Internet infrastructures. Across four experimental scenarios, the results demonstrate that EcoRouting can achieve substantial reductions in carbon emissions, up to 47.1% in multi-city topologies, while maintaining network operability. However, these environmental benefits often come at the cost of increased latency, which in some cases reached more than 90%.

The findings highlight that EcoRouting's performance is highly dependent on both network topology and the variability of carbon intensity across the grid. Dense and diverse topologies, as well as environments with significant temporal fluctuations in emissions, offer greater opportunities for carbon optimization. In contrast, homogeneous or topology constrained networks yield limited benefits, as routing alternatives are restricted.

Future research should focus on integrating adaptive mechanisms that dynamically balance carbon reduction and latency in real-time, potentially leveraging machine learning models for predictive routing. Additionally, extending the evaluation to real-world network measurements, beyond synthetic simulations, would provide stronger evidence of EcoRouting's operational viability in production scale Internet infrastructures.

REFERENCES

[1] M. Wang, Z. Zhu, Y. Wang, and S. Xie, "Energy Efficient and Highly Reliable Geographic Routing Based on Link Detection and Node Collaborative Scheduling in WSN," Sensors, vol. 24, no. 11, Art. 3263, Jun. 2024. [Online]. Available: https://www.mdpi.com/1424-8220/24/11/3263

[2] K. Haseeb, F. A. Alzahrani, M. Siraj, Z. Ullah, and J. Lloret, "Energy Aware Next Generation Mobile Routing Chains with Fog Computing for

Emerging Applications," Electronics, vol. 12, no. 3, Art. 574, Mar. 2023. [Online]. Available: https://www.mdpi.com/2079-9292/12/3/574

[3] S. Tabaeiaghdaei, S. Scherrer, J. Kwon, and A. Perrig, "Carbon Intelligent Global Routing in Path Aware Networks," in Proc. ACM eEnergy, Jun. 2023, pp. 144–158. [Online]. Available: https://dl.acm.org/doi/10.1145/3575813.3595192

[4] S. Qi, H. Moore, H. Hogade, N. Milojicic, C. P. Bash, and S. Pasricha, "CASA: A framework for SLO- and carbon-aware autoscaling and scheduling in serverless cloud computing," in Proc. IEEE 15th Int. Green and Sustainable Computing Conf. (IGSC), Oct. 2024, pp. 1–8. doi: 10.1109/IGSC64514.2024.00010.

[5] Y. Liu, H. Zhang, and X. Zhao, "Realizing the Carbon Aware Service Provision in ICT System," IEEE Transactions on Network and Service Management, 2024. DOI: 10.1109/TNSM.2024.3385484

[6] D. Otten, S. Neuner, N. Aschenbruck, and T. Schüller, "On modeling the power consumption of a backbone network," in Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops), Rome, Italy, 2023, pp. 1842–1847, doi: 10.1109/ICCWorkshops57953.2023.10283615.

[7] S. El Zahr, P. Gunning, and N. Zilberman, "Exploring the Benefits of Carbon Aware Routing," Proc. ACM on Networking, vol. 1, CoNEXT3, Nov. 2023. [Online]. Available: https://dl.acm.org/doi/10.1145/3629165

[8] Electricity Maps (2024). (INDONESIA) Carbon Intensity Data (Version July 3, 2025). Electricity Maps. https://www.electricitymaps.com

[9] M. Lin and L. Schaposnik, "A carbon-aware ant colony system for the sustainable generalized traveling salesman problem," Scientific Reports, vol. 15, no. 21928, 2025. doi: 10.1038/s41598-025-08276-9.

[10] M. R. Poornima, H. S. Vimala, and J. Shreyas, "Holistic survey on energy aware routing techniques for IoT applications," Journal of Network and Computer Applications, vol. 213, p. 103584, 2023. doi: 10.1016/j.jnca.2023.103584.

[11] S. Tache, E. Benkhelifa, and M. A. Ferrag, "SDN-based traffic engineering and routing optimization: A comprehensive review," Applied Sciences, vol. 14, no. 14, p. 5967, 2024. doi: 10.3390/app14145967.

[12] S. Tache, L. Abderrahmane, and M. A. Ferrag, "Metaheuristic optimization techniques for software-defined networks: Trends and challenges," Journal of Cloud Computing, vol. 13, no. 1, p. 73, 2024. doi: 10.1186/s13677-024-00603-1.

[13] S. Hosseini, S. Farahani, and A. M. Rahmani, "Fuzzy metaheuristic-based energy-aware routing for SDN-enabled IoT networks," International Journal of Sensor Networks, vol. 41, no. 4, pp. 265–279, 2023. doi: 10.1504/IJSNET.2023.132543.

[14] K. Haseeb, I. U. Din, A. Almogren, M. K. Khan, and H. A. Khattak, "Energy-aware routing in fog-enabled mobile networks for sustainable IoT applications," Electronics, vol. 12, no. 3, p. 574, 2023. doi: 10.3390/electronics12030574.

[15] D. Hong, Y. Liang, H. Sun, et al., "DFRDRL: A dynamic-flow reliable and dynamic routing algorithm based on reinforcement learning for SDN," Journal of Big Data, vol. 11, no. 102, 2024. doi: 10.1186/s40537-024-00938-6.

[16] H. Han, J. Tang, and Z. Jing, "Wireless sensor network routing optimization based on improved ant colony algorithm in the Internet of Things," Heliyon, vol. 10, no. 1, e23577, 2024. doi: 10.1016/j.heliyon.2023.e23577.

[17] A. Radovanović, J. Cho, M. Hajiesmaili, D. Irwin, P. Shenoy, and R. K. Sitaraman, "Carbon-aware computing for datacenters," IEEE Transactions on Power Systems, vol. 38, no. 2, pp. 1270–1280, Mar. 2023. doi: 10.1109/TPWRS.2022.3218495.

[18] H. H. Xu and M. van der Schaar, "Effective bandwidth estimation and admission control for video communication in IEEE 802.11e EDCA WLANs," in Proc. IEEE Int. Symp. Modeling, Analysis & Simulation of Computer and Telecommunication Systems (MASCOTS), Aug. 2013, pp. 155–164. [Online]. Available: https://henryhxu.github.io/share/mascots13.pdf.

[19] Y. Jiang, J. Zhu, and K. Yang, "Environment-aware adaptive reinforcement learning-based routing for vehicular ad hoc networks," Sensors, vol. 24, no. 1, p. 40, 2024. doi: 10.3390/s24010040.

[20] T. A. Gamage and I. Perera, "Optimizing energy efficient cloud architectures for edge computing: A comprehensive review," International Journal of Advanced Computer Science and Applications, vol. 15, no. 11, pp. 637–648, 2024. doi: 10.14569/IJACSA.2024.0151161.

[21] D. F. de F. Medeiros, C. P. de Souza, F. B. S. de Carvalho, and W. T. Araújo Lopes, "Energy saving routing protocols for smart cities using LoRa-based WSNs," Energies, vol. 15, no. 19, Art. 7382, Oct. 2022. doi: 10.3390/en15197382.

[22] M. Khalil, F. Al-Turjman, and S. Alturjman, "Carbon footprint reduction in next-generation Internet routing: A multi-objective approach," Journal of Network and Computer Applications, vol. 214, p. 103530, 2023. doi: 10.1016/j.jnca.2023.103530.

[23] T. Nguyen, Q. Pham, and J. Kim, "Time-aware optimization for sustainable network operations," IEEE Access, vol. 12, pp. 14782–14795, 2024. doi: 10.1109/ACCESS.2024.3356102.

[24] R. Das, A. Kumar, and S. Saha, "Energy-aware routing strategies for sustainable Internet infrastructures," IEEE Transactions on Green Communications and Networking, vol. 7, no. 4, pp. 2156–2170, 2023. doi: 10.1109/TGCN.2023.3261543.

[25] E. Morales, L. Zhang, and Y. Chen, "SDN-enabled carbon-aware routing in large-scale networks," Computer Networks, vol. 243, p. 110625, 2024. doi: 10.1016/j.comnet.2024.110625.

Exploring the Future Research Agenda for Health Applications Adoption: A Systematic Literature Review

Rahmat Fauzi[1], Adhistya Erna Permanasari[2]*, Silmi Fauziati[3]
Department of Electrical Engineering and Information Technology, Universitas Gadjah Mada, Yogyakarta, Indonesia[1, 2, 3]
School of Industrial Engineering and System, Telkom University, Bandung, Indonesia[1]

***Abstract*—The healthcare sector is experiencing rapid digital transformation, marked by the growing popularity of mobile health (mHealth) and eHealth applications for various health-related purposes. However, despite their potential, the adoption of health applications remains inconsistent due to varying influencing factors. Previous reviews often focused on specific populations or limited frameworks, leaving a gap for a comprehensive synthesis. This study aims to systematically review and consolidate the current understanding of the factors affecting user adoption behavior in health applications. Following the Preferred Reporting Items for Systematic Review and Meta-Analyses (PRISMA) guidelines, a comprehensive literature search was conducted to identify relevant studies between 2016 and 2025. A total of 79 primary studies were analyzed to explore the theoretical model, variables, and emerging trends in health applications. The Technology Acceptance Model (TAM) and Unified Theory of Acceptance and Use of Technology (UTAUT) models are the most widely used models by researchers. Beyond these core frameworks, researchers have proposed extended constructs such as psychological factors, health literacy, regulator readiness, security concerns, and infrastructure limitations. This review highlights the need for more inclusive, cross-cultural, and mixed-method research, particularly focusing on underrepresented populations such as rural users, the elderly, and low-literacy groups. These findings offer valuable insight to inform the design of future models and support the development of more effective, context-aware, and user-centered health technologies.**

***Keywords*—Systematic literature review; health application adoption; health technology; user behavior; technology acceptance; TAM; UTAUT**

I. Introduction

In the digital era, internet usage has grown significantly, supported by the rapid advancement of information and communication technology (ICT), including in the healthcare sector. The development of increasingly sophisticated and innovative digital health applications has become a preferred choice for the community because it offers facilities and services that are useful in everyday life. Digital health applications are designed to enhance health awareness, promote healthy lifestyles, lifestyle management, provide access to medical services, monitor medical conditions, interaction between doctors, health professionals and patients. In general, Digital health applications offer customers digital health information services through mobile applications such as Android and iOS or web-based applications. According to the World Health Organization (WHO), eHealth refers to the effective and secure utilization of information technology within the healthcare industry, encompassing domains such as healthcare services, disease monitoring, health-related literature, education, research, and knowledge dissemination [1].

Although health applications have the potential and help improve people's health, their use still faces numerous challenges [2]. People, especially in developing countries, still do not intend to use or tend to rarely use health applications. This raises the question of which factors influence sustained use of health applications. Prior studies on mobile health (mHealth), such as the study by Jacob et al. (2022) [3], have investigated socio-technical factors influencing adoption, including technical barriers, health literacy, social influence, and highlighted challenges such as patient resistance and ethical concerns. However, while their study focused specifically on patient perspectives, this review emphasizes broader technology adoption models—such as the Technology Acceptance Model (TAM) and Unified Theory of Acceptance and Use of Technology (UTAUT)—to identify behavioral, social, technical, and security-related factors that collectively affect adoption. Unlike previous research, this study also considers multiple stakeholders, including users, developers, and policy factors—and focuses on general health applications rather than disease-specific tools, thus offering a more comprehensive view of the adoption ecosystem.

This study aims to systematically review and consolidate the factors influencing health application adoption, identify the most widely applied theoretical models and their extensions, and categorize adoption determinants into eight domains: technical, psychological, security and trust, cognitive/literacy, health attitude, social, regulatory, and economic. By doing so, the study provides a comprehensive evidence base to inform the design of inclusive, context-aware, and user-centered health technologies.

The remainder of this study is organized as follows: Section II outlines related work to establish the relevance and contribution of this study. Section III describes the methodology, including the PRISMA-based SLR process. Section IV presents the results derived from the research questions. Section V discusses the findings, highlights research gaps, and proposes a future research agenda. Finally, Section VI concludes with implications for researchers, developers, and policymakers.

*Corresponding Author

II. Related Work

Research on health application adoption has been explored through a range of systematic reviews, each addressing specific technologies, context, or user groups. For example, Aljohani and Chandran [4] systematically reviewed 22 studies on mHealth adoption in developing countries, showing the predominance of TAM but focusing narrowly on patient perspectives and offering limited integration of broader socio-cultural or policy-related factors. Similarly, Ghiwaa et al [5] analyzed 21 studies on telemedicine adoption between 2012 and 2023, identifying TAM and UTAUT as the most frequently applied frameworks but restricting the scope telemedicine delivery contexts and overlooking adoption determinants beyond service providers and patients. In another direction, Ikwunne et al. [6] reviewed on mHealth design processes, synthesized 32 studies and proposed a 16-item checklist to enhance user engagement. However, while valuable for guiding design practices, it did not address adoption determinants systematically across theoretical models.

Beyond these works, reviews in other domains have focused on narrower technological contexts. Jacob et al. [3] examined the adoption of mobile health applications in chronic disease management, emphasizing usability, engagement, and security factors, but its disease-specific focus limited generalizability. Likewise, Al-rawashdeh et al. [7] investigated adoption factors for wearable devices, highlighting perceived usefulness, trust, and cost as critical, but did not integrate a cross-framework perspective or broader population groups. Collectively, these prior reviews provide useful but fragmented insights into adoption, as they tend to emphasize specific technologies, health conditions, or populations, and rarely combine multi-stakeholder viewpoints.

In contrast, the present study contributes a broader and more comprehensive synthesis. By systematically analyzing 79 primary studies published between 2016 and 2025, it integrates multiple theoretical models—including TAM, UTAUT, Theory of Planned Behavior (TPB), Information System (IS) Success Model, and others—while categorizing determinants into eight thematic domains (technical, psychological, security and trust, cognitive/literacy, health attitude, social, regulatory, and economic). Furthermore, this study incorporates a multi-stakeholder perspective encompassing users, developers, and policymakers, and pays explicit attention to underrepresented groups such as rural, elderly, and low-literacy populations. This approach not only consolidates existing findings but also highlight emerging post-COVID-19 trends, thereby offering a timely and holistic contribution to the health application adoption literature.

III. Methodology

A systematic approach for reviewing the literature on health application user behavior is chosen. Systematic literature review (SLR) is now the most widely used review method in the field of information systems. Systematic literature review (SLR), commonly known as systematic review, involves the process of locating, assessing, and analyzing all relevant research studies associated with a specific research topic, question, or phenomenon of interest [8]. The purpose of this research is to explore these questions by conducting a comprehensive examination of prior relevant research and integrating it with current theories using a systematic literature review (SLR) approach. As illustrated in Fig. 1, the Kitchenham SLR method consists of three processes: planning, conducting, and reporting [8].

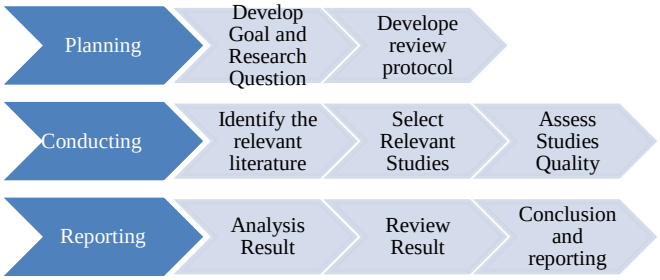


Fig. 1. Systematic Literature Review (SLR) method process.

Undoubtedly, the three key stages of a Systematic Literature Review (SLR) comprise planning, conducting, and reporting the literature review, as depicted in Fig. 1. In the initial phase (Step 1), the requirements for a systematic review are established. In the opening section of this work, the objectives of conducting the literature review were elucidated. Following this, existing systematic reviews on consumer approval of health application activity are identified and evaluated. The purpose of the review process is to provide direction and minimize the likelihood of researcher bias (Step 2). This entails defining the study selection criteria, formulating research questions, determining the search method, assessing quality, and ultimately, extracting and synthesizing data (Step 3).

A. SLR Planning Process

The purpose of specifying the Research Questions (RQ) is to maintain the focus of the review. These questions are formulated using the Population, Intervention, Comparison, Outcomes, and Context (PICOC) criteria, as outlined by Kitchenham and Charters (2007). The structure of the research questions based on the PICOC criteria is presented in Table I.

TABLE I Summary of Picoc

Structure	Scope and Coverage
Population	Digital Health, eHealth, Mobile Health, mHealth, Health Application
Intervention	Model, Method, Acceptance, Readiness, Adoption
Comparison	TAM, UTAUT, The Technology-Organization-Environment (TOE), The Theory of Planned Behavior (TPB), The Health Belief Model (HBM)
Outcome	Successful Adoption, Model Adoption, Technology Behavior
Context	Studies in Health Industry, Software Development, and academia Research

The Research Questions (RQ) and motivation that this literature review addresses are presented in a tabular format as Table II.

TABLE II RESEARCH QUESTIONS ON LITERATURE REVIEW

ID	Research Questions	Motivation
RQ1	Which journal is the most significant publication for user adoption of health applications?	Identify the most significant journal in the model adoption of health application.
RQ2	What kind of theory adoption approach is used most often for user behavior on health applications?	Identify research topics and trends in the model adoption of health applications.
RQ3	What are the factors or constructs that mostly influence user behavior in health application adoption?	Identify the most used factor for model adoption of health applications.

B. SLR Conducting Process

In the process of creating SLR, it is necessary to search and identify all original published articles related to user behavior adoption in health applications. Before beginning the search library, it is vital to choose a set of databases to enhance the finding of highly relevant articles. Science Direct, Springer, Taylor & Francis, IEEE, and Emerald are the most comprehensive lists used for literature search. It is all recorded by the Scopus Database. As displayed in Fig. 2, this review has been conducted in accordance with the Preferred Reporting Items for Systematic Reviews and Meta-Analysis (PRISMA) guidelines [9], [10]. Employing the PRISMA framework demonstrates the progression of information through the various stages of the review [9], [10]. Additionally, it illustrates the quantity of articles that were identified, included, and excluded, along with the rationale for excluding certain articles. The methods used to identify and gather the relevant studies in this review encompassed multiple phases, including defining inclusion and exclusion criteria, determining the sources and digital databases, specifying search strategies, and evaluating the retrieved studies.

The search for pertinent studies was executed using databases that were specifically selected for this objective. In October 2023, a search was carried out to gather research papers published from 2016 to May 2025, covering a ten-year duration. A search strategy was devised using specific keywords that are outlined in Table III. The initial search results yielded a total of 736 studies, as depicted in Fig. 2. The inclusion and exclusion criteria were then applied, and the refinement stages as per the PRISMA were followed. The analysis of the collected studies was performed by the first and third authors of this study, who analyzed each article independently. Any discrepancies in the analysis of the studies between the two authors were resolved through discussion and further review of the disputed studies. As a result, a total of 79 studies were deemed to be valid and were included in the analysis.

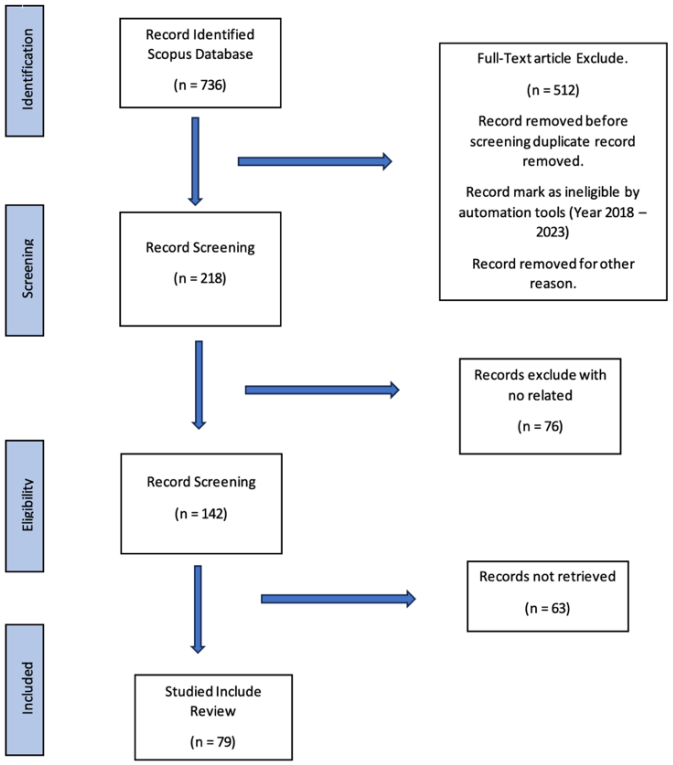


Fig. 2. PRISMA flow diagram.

The inclusion and exclusion criteria are established to determine the selection rules for studies prior to the analysis phase, as indicated in Table III. These criteria are critical for determining the validity of studies to be included in the analysis and ensuring consistency among the reviewed studies.

TABLE III INCLUSION AND EXCLUSION CRITERIA

Criterion	Criterion Dimension
Inclusions	1. Authentic study results were selected only those were published in Scopus journals. 2. Articles that include the terms "Health" and "Model Adoption" in the title, abstract, or keywords. 3. Articles that include the terms "Health" and "Model Acceptance" in the title, abstract, or keywords. 4. Articles that include the terms "Health" and "Model Acceptance" in the title, abstract, or keywords. 5. Article between 2016 and 2025
Exclusions	1. Studies not written in English 2. The paper is recorded in Scopus only in Q1, Q2, Q3, and Q4. 3. The present research focuses on applying technology acceptance or adoption, but not in the context of healthcare (e.g. banking, e-learning, e-commerce). 4. The topic is related to the User perspective. Not Professional healthcare or health worker. 5. The topic focuses on health applications, either mobile or website-based. Not Wearable Device or IoT healthcare.

1) Data collection: The initial phase of the study involved selecting 79 primary studies from 736 publication for analysis. All these studies were thoroughly examined to ensure they met the inclusion and exclusion criteria, as well as to assess their quality, relevance to the research questions, and similarity to one another. Any duplicate studies by the same authors that were published in different journals were removed, leaving a total of 79 primary studies. The complete list of selected studies can be found in the final section of this study. In terms of distribution, Q1 and Q2 dominate the distribution of search results. Data distribution was 47 % Q1, such as Journals from BMC Medical Informatics and Decision Making, Digital Health, Technology in Society, IEEE Access, Journal of Migration and Health, International Journal of Medical Informatics, JMIR mHealth and uHealth, JMIR Public Health and Surveillance, Journal of Biomedical Informatics, etc. Besides, 39 % data distribution was Q2 consist of Journals such as Future Internet, Informatics in Medicine Unlocked, Informatics in Medicine Unlocked, International Journal of Telemedicine and Applications, Healthcare, Health Care for Women International, Transforming Environment: People, Process and Policy, Journal of Computer Information Systems, etc. In contrast, the distribution of Q3 and Q4 data was not as large as the previous data. Q3 contributed 11% which was spread from journals such as TEM Journal, International Journal of Pharmaceutical and Healthcare Marketing, Kybernetes, Journal of Systems and Management Science, and International Journal of Pharmaceutical and Healthcare Marketing. Lastly, 3% was contributed from Q4, where two papers from the journal Asia Pacific Journal of Health Management completed the distribution of publications. Fig. 3 illustrates the distribution quartile of journal articles.

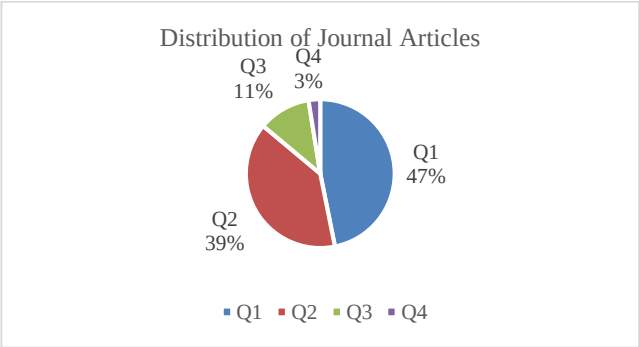


Fig. 3. Quartile distribution of journal articles.

2) Data extraction: The required data to address the research questions outlined in this review were obtained from the selected primary studies. A customized data extraction form was created for each of the 79 chosen primary studies, with a focus on obtaining the necessary information to answer the research questions. These research questions include RQ1: Significant Journal Publications, RQ2: Theory Adoption Approach Used, RQ3: Factors Influencing Adoption.

All collected data adhered to the quality standards set by the study and was utilized to address the research questions. After extraction, the relevant information from each study was compiled into a single Excel file. This file was then used for manual sorting and categorization based on the three research questions. Journals were grouped according to their significant contributions, the theoretical models employed, the factors they identified influencing adoption, any proposed modifications to existing adoption models, and the key challenges and opportunities highlighted in each study. The manual sorting process allowed for a structured approach to analyzing the studies and identifying key patterns and trends.

Additionally, a keyword mapping exercise was conducted using VOS Viewer for the content analysis of research papers related to health app trends, as shown in Fig. 4. The VOS Viewer analysis identified commonly recurring terms such as "Human", "Patient", "Adoption", "Healthcare", and "mHealth". This analysis helped to clarify the broader research focus and trends, complementing the detailed sorting process carried out in Excel. The combination of VOS Viewer for keyword mapping and Excel for manual data sorting enabled a comprehensive understanding of how the research landscape has evolved in health application adoption. The findings from both the Excel sorting and VOS Viewer analysis revealed significant trends related to business models for using health applications, the integration of health applications to monitor health history during the COVID-19 pandemic, and the utilization of decision support systems, artificial intelligence, machine learning, and big data. These findings were sourced from the Scopus database, offering valuable insights for further exploration of health app adoption trends.

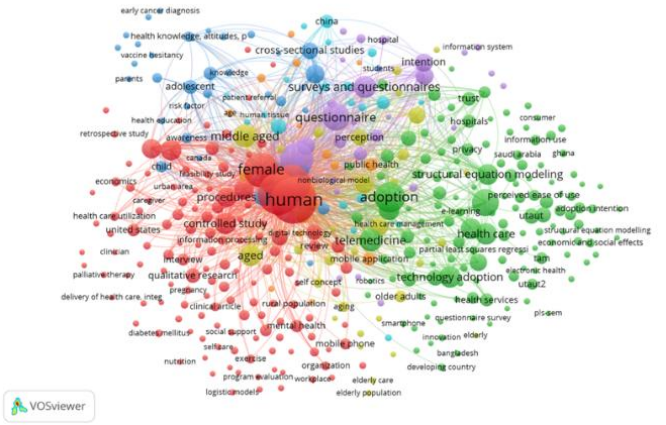


Fig. 4. VOS Viewer's content analysis.

IV. Results

A. RQ1 : The Most Significant Journal Publications

Based on the literature review, 79 primary studies were identified that examined the performance of user behavioral health applications. The data shows a significant increase in research on user behavior in adopting health applications over the past ten years, as evidenced by several published studies, especially since the COVID-19 pandemic in 2020. Fig. 5 depicts research on user behavior and acceptance of health applications. This graph can be used to show that interest in the topic is

increasing year by year. This graph can also be used to show that the topic is a popular topic and interesting to research. The COVID-19 pandemic led to a significant surge in research on the use of health applications from 2019 to 2020. In the subsequent four years, there has been a corresponding increase in studies related to the adoption of health application, aligning with the heightened public consciousness of health issues.

The following Table IV displays the most significant user behaviors in health app acceptance journals, as demonstrated in primary studies that were selected for this purpose. Additionally, Table IV shows the most significant reader behaviors in health application journals, along with their Scimago Journal Rank (SJR) scores and Q categories. The ranking of journal articles is based on their SJR values, which are listed in the table in descending order.

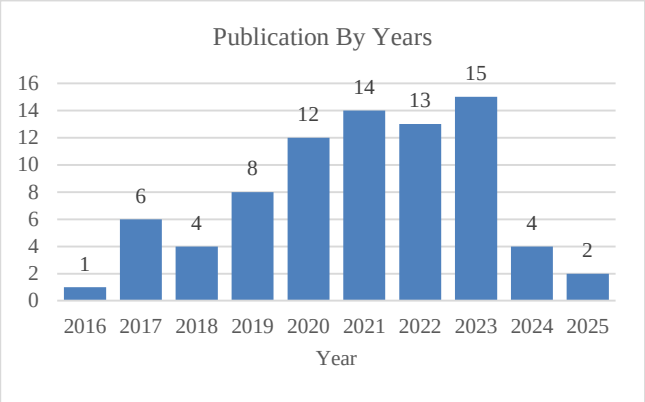


Fig. 5. Article distribution by years from selected papers.

TABLE IV THE MOST SIGNIFICANT DISTRIBUTION SCIMAGO JOURNAL RANK (SJR) OF SELECTED JOURNALS

No	Journal Name	Publisher	SJR	Q in Category
1	International Journal of Information Management journal	Elsevier	4,91	Q1 in Information System
2	Journal of Business Research	Elsevier	2,9	Q1 in Marketing
3	Technological Forecasting and Social Change	Elsevier	2,64	Q1 in Management technology
4	Journal of Medical Internet Research	JMIR Publication	1,99	Q1 in Health Informatics
5	JMIR Public Health and Surveillance	JMIR Publication	1,55	Q1 in Health Informatics
6	JMIR mHealth and uHealth	JMIR Publication	1,51	Q1 in Health Informatics
7	Technology in Society	Elsevier	1,491	Q1 in Business and International Management
8	Journal of Migration and Health	Elsevier	1,45	Q1 in Health (Social Science)
9	Telemedicine Journal and e-Health	Mary Ann Liebert, Inc.	1,24	Q1 in Health Informatics
10	Telemedicine and E-health	Mary Ann Liebert, Inc.	1,237	Q1 in Health Informatics

Table V shows the number of articles published in each journal in the field of information in the health sector. International Journal of Environmental Research and Public Health and Informatics in Medicine Unlocked were the journals most frequently published with four articles each, followed by the journal Technology in Society with three articles, IEEE Access, International Journal of Pharmaceutical and Healthcare Marketing, and Asia Pacific Journal of Health Management with two articles. Table V shows that the field of health application adoption research is active, developing and has increased in the last five years, and there are several journals that are important sources of information for researchers.

TABLE V THE MOST CONTRIBUTED PAPER

No	Journal Name	Quartile	Amount	Source
1	International Journal of Environmental Research and Public Health	Q2	6	[11], [12], [13], [14], [15], [16]
2	Informatics in Medicine Unlocked	Q2	4	[2], [17], [18], [19]
3	International Journal of Medical Informatics	Q1	3	[20], [21], [22]
4	JMIR mHealth and uHealth	Q1	3	[23], [24], [25]
5	Technology in Society	Q1	3	[26], [27], [28]
6	Informatics for Health and Social Care	Q2	3	[29], [30]
7	International Journal of Pharmaceutical and Healthcare Marketing	Q3	3	[31], [32], [33]
8	IEEE Access	Q1	2	[34], [35]
9	International Journal of Pharmaceutical and Healthcare Marketing	Q3	2	[31], [32]
10	Asia Pacific Journal of Health Management	Q4	2	[36], [37]

B. RQ2: The Most Theoretical Approach used

The presented bar chart in Fig. 6 exhibits the distribution of models utilized in the development of health and fitness applications. It is evident that the UTAUT (Unified Theory of Acceptance and Use of Technology) & Extend model holds the predominant position, with 32 findings derived from all applications employing this model. The TAM & Extend model ranks second, with 31 findings originating from all applications utilizing this model. The Information System (IS) Success Model ranks third, appearing in four studies, indicating a moderate but meaningful influence, particularly in contexts where system quality, information quality, and user satisfaction are being evaluated. The Technology Readiness Model was applied in three studies, suggesting an emerging interest in assessing users' predisposition toward new technologies. Models such as the Theory of Planned Behavior (TPB) and Social Cognitive Theory (SCT) were each used in two studies,

often to explore behavior intentions and the influence of perceived control or self-efficacy. Meanwhile, less frequently used model—including the Organizational Support Theory, Diffusion of Innovation Theory (DOI), Technology-Organizational-Environment (TOE) Framework, Health Belief Model, and Interview each appeared in only one study. These limited occurrences suggest that while they may offer unique perspectives, they are not yet widely adopted in the health application domain.

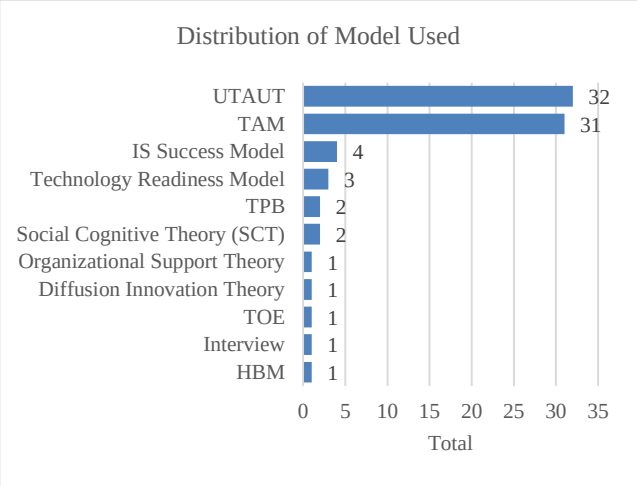


Fig. 6. Distribution of the model used to analyse user behavior in health application adoption.

C. RQ3: The Most Factor or Construct that Influence User

The adoption of health application products and services by users can be influenced by various factors. Identifying and understanding these factors, as well as their impact on the acceptance of health applications, is crucial for researchers in formulating strategies to promote the adoption of health applications and encourage positive user behavior. The classification chart offers a comprehensive summary of non-core variables influencing user adoption of health applications, grouped into eight thematic categories: technical, security and trust, cognitive/literacy, innovativeness, psychological, health attitude, social, regulatory and economic. These factors go beyond core constructs of models like TAM, UTAUT, TPB, or IS Success Model and allow for a deeper understanding of the behavioral, contextual, and environmental elements affecting adoption as illustrated in Fig. 7.

The technical category, which comprises 22.1% of all variables, includes aspects such as compatibility [21], [38], [39], [40], [41], [42], application quality [43], and technology accuracy [42]. These features are essential to ensuring that user experience reliable and responsive systems, which in turn support continued use and engagement. Psychological factors contribute 20.6% of the variables and include constructs such as technology anxiety [11], [22], [26], [28], [44], [45], [46], [47], [48], fatigue [49], perceived enjoyment [32], [37], [50], [51], and intolerance of uncertainty [49]. These variables highlight the emotional and cognitive processes that shape user confidence, resistance, or willingness to adopt digital health platforms. The security and trust category (8.8%) contains variables such as perceived privacy [11], [14], [16], [20], [23], [25], [27], [28], [29], [32], [42], [43], [52], [53], [54], [55], trust [12], [17], [20], [23], [25], [26], [27], [28], [29], [31], [32], [36], [39], [47], [53], [55], [56], [57], [58] , and perceived risk [12], [25], [26], [28], [47], [55], [56], [58], [59]. These elements indicate that without a strong foundation of user trust –especially concerning data protection—adoption is likely to falter, regardless of system quality.

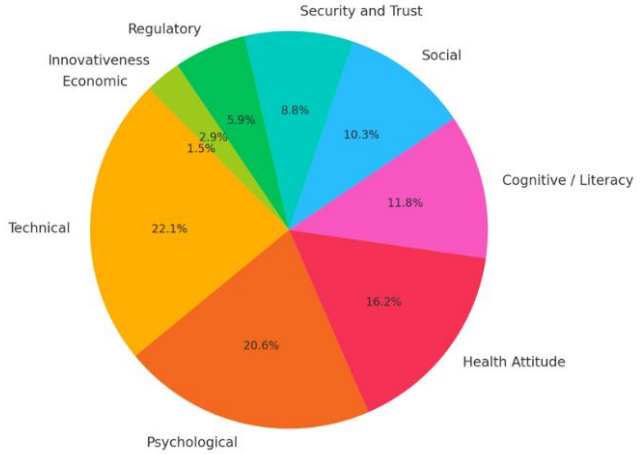


Fig. 7. Distribution of most factors or variables that influence users in health acceptance.

In the cognitive/literacy category (11.8%), factors such as health literacy, IT literacy, and awareness reflect the role of user knowledge in facilitating or hindering adoption. Individuals with a limited understanding of technology or health content may require additional support to engage with applications effectively. Health attitude factors make up 16.2% encompassing health consciousness [12], [36], [42], [60], [61], [62], disease threat perception [43], [47], and personal health status [63]. These beliefs directly influence health behavior, particularly when users perceive a high level of personal risk or benefit. Social factors account for 10.3% of variables and include community support [15], word of mouth [58], and social values [15]. This category emphasizes the role of social environments and peer influence in encouraging technology use, especially in tightly knit or communal populations. The regulatory category (5.9%) includes management support [21], legal concern [25], and user involvement in application implementation [21]. These institutional-level factors highlight that, beyond individual behavior, organizational readiness and policy alignment are key to scaling adoption. Finally, economic considerations account for 1.5% of variables, with affordability [17], [64] being particularly relevant in developing country contexts. The cost burden can be a critical barrier, especially when users perceive that value as insufficient compared to expenses.

V. Analysis, Discussions, And Future Research Agenda

A. Analysis of Users' Acceptance and Behavior in Health Application

The literature collected in this review reveals significant developments and future potential in health adoption research. In recent years, the volume of studies exploring adoption from

users, business, and developer perspectives has increased substantially, reflecting growing interest in understanding the multidimensional dynamics of digital health implementation. Analyzing the VOSviewer data in Fig. 8 illustrates the quantitative data from the collected research literature. Fig. 8 displays a dense and interconnected network of keywords, where each node represents a recurring theme or topic within the literature, and the links illustrate co-occurrence relationships. Based on the analysis, it shows that there is still a growing awareness and potential for research in the topic of health technology adoption, with various factors such as knowledge, attitudes, and behavior of the public. The overlay visualization further incorporates a temporal dimension, with node colors ranging from purple to yellow (indicating 2020 to 2021). The central position of keywords such as "human", "health care", "telemedicine", "mhealth", and "technology acceptance" indicates that these topics have served as conceptual anchors throughout the reviewed period.

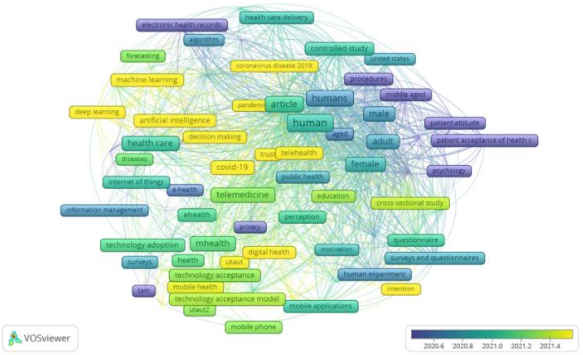


Fig. 8. VOSviewer overlay.

The overlay also shows a noticeable shift in research focus, particularly following the emergence of the COVID-19 pandemic. Keywords such as "covid-19", "telehealth", "machine learning", and "artificial intelligence" appear in brighter hues, reflecting their more recent prominence. This suggests that the global health crisis accelerated the public's awareness of digital health solutions and intensified scholarly interest in the adoption of mobile and AI-enabled health technologies. Accordingly, research on user acceptance of health services has become increasingly prominent, especially as the pandemic exposed both the potential and the urgency of scalable digital interventions. Beyond the influence of COVID-19, the visualization also highlights important sociotechnical factors—such as education, privacy, trust, intention, and perception—that continue to play a critical role in health technology adoption. Social, economic, and demographic factors, including age, gender, and patient attitude, also appear prominently and indicate a growing concern with inclusivity and user diversity. This suggests that future research must be attentive not only to technological readiness, but also to user-specific needs, access inequalities and cultural factors.

Overall, the network shows a research landscape that is theoretically grounded, empirically rich, and methodologically diverse. It offers valuable insights for researchers, health professionals, and policymakers, emphasizing the need to refine behavioral models, integrate intelligence technologies, and design user-centered health applications that are responsive to both global emergencies and long-term public health goals.

On the other hand, this research topic is also on the rise as evidenced by its distribution in several countries. Table VI depicts the distribution of the number of publications based on countries related to the topic of user behavior and health app adoption. The amount indicates that China has the highest number of study cases with 13 publications, followed by Bangladesh and Indonesia with 12 and 8. Saudi Arabia and India are tied with a collection of 6 studies each. According to the data from this study, the topic of user behavior and health app adoption is gaining popularity in developing countries.

TABLE VI DISTRIBUTION OF RESEARCH BY COUNTRY

Name of Country	Amount	References
China	13	[15], [16], [25], [30], [42], [47], [48], [54], [60], [63], [65], [66], [67]
Bangladesh	12	[22], [27], [29], [33], [38], [46], [51], [61], [62], [68], [69], [70]
Indonesia	8	[17], [21], [39], [50], [71], [72], [73], [74]
Saudi Arabia	6	[13], [14], [35], [40], [75], [76]
India	6	[20], [26], [36], [64], [77], [78]
Pakistan	4	[28], [45], [53], [79]
Germany	3	[23], [49], [59]
Portugal	2	[41], [80]
South Korea	2	[11], [37]
Malaysia	2	[81], [82]
Iran	2	[2], [83]
Other countries	19	[12], [18], [24], [31], [32], [34], [43], [44], [52], [56], [57], [58], [84], [85], [86], [87], [88], [89], [90]

B. Discussions and Future Research Agenda

Based on the review of 79 selected manuscripts, several recurring challenges and thematic gaps were identified in the current landscape of health application adoption research. Addressing these gaps is essential to ensure that digital health technologies are inclusive, sustainable, and effective across diverse contexts. One of the persistent issues is demographic bias and inclusivity in health application adoption [12], [14], [20]. Many studies remain focused on younger, urban, well-educated populations with high digital literacy. While these groups represent important early adopters, such a narrow sampling fails to capture the boarder realities of health technology users. Marginalized populations—such as rural communities, older adults, individuals with disabilities and socioeconomically disadvantaged groups—are often underrepresented or excluded altogether. Similarly, generational or gender-based differences are rarely explored in depth. Thus, future research must therefore move beyond convenience sampling toward more purposeful designs, such as stratified sampling or community-based participatory approaches, to ensure the inclusion of underrepresented groups.

Closely related is the challenge of health and IT literacy, which remains as a critical barrier to effective and sustained use of mobile health (mHealth) applications. Users with limited health knowledge, low digital skills, or minimal formal

education often struggle to navigate app functionalities or interpret health information accurately [2], [34], [44]. This issue is particularly acute among groups already facing systemic barriers to healthcare access, such as those from rural area or low-income background. To address this, researchers and developers must prioritize accessible and inclusive principles, including simplified interface, multilingual and voice-assisted instructions, intuitive navigation, and visual cues that require minimal technical knowledge. Complementary efforts, such as community-based digital literacy training, can empower users with limited experience to build confidence and competence in using application health.

Another recurring limitation is the restricted geographical and contextual scope of existing research. Many adoption studies have been conducted within single countries—frequently in urban or institutional settings. This limitation reduces external validity and raises questions about whether theoretical models can be generalized across regions with differing health systems, infrastructure, and cultural norms [14], [26], [27]. Adoption behaviors are not universal but are shaped by contextual realities such as policy framework, economic conditions, and cultural attitudes toward health and technology. To address this, future research should expand into cross-country, and cultural comparative designs. Approaches such as structural equation modelling across diverse populations, meta-analyses, and multinational collaborations could help identify both universal and context-specific determinants of adoption.

The challenge of cost and infrastructure readiness also remains underexplored. Although digital health applications are often promoted as scalable and cost-efficient, the reality of their implementation is often constrained by financial burden and technological limitations [44],[50],[51]. Developing, deploying, and maintaining health applications requires significant investment not only in software, but also in cellular network support, data security protocols, and user support systems. These hidden costs are often not accounted for in studies that focus solely on user-side factors. Addressing these issues requires a strategic plan. First, researchers and developers should prioritize the creation of lightweight, device-independent, offline-capable applications optimized for low-bandwidth environments, while policymakers should integrate digital health into national healthcare plans ensuring alignment with existing systems and long-term financial planning. Public-private partnerships, government subsidies, and donor-funded initiatives may also play a key role in supporting infrastructure, especially in rural area.

The issue of security and privacy remains a critical deterrent to adoption [64], [71], [73]. Although technical safeguard such as encryption and authentication have improved, user perception of trust, transparency, and control are equally influential in shaping behavior. Many applications lack clear consent mechanisms, easy-to-use privacy settings, or transparent data-sharing policies. Inconsistent adherence to international standards amplifies hesitancy, especially in countries with weak regulatory oversight. Future studies should not only advance technical solutions through privacy-by-design approaches but also investigate behavioral and social determinants of trust. Educating users, simplifying privacy tools, and aligning with robust regulatory frameworks are crucial for building confidence in health applications.

Policy and government support also emerged as a foundational yet underdeveloped factor. Despite the growing interest in digital health, many countries lack coherent digital health strategies, legal frameworks, and institutional coordination to support health application adoption [64], [65], [70]. Without government endorsement or integration into formal health systems, health applications often remain fragmented, underfunded, and poorly interoperable. This leads to poor interoperability, weak sustainability, and low public trust. Moving forward, the government must invest in digital health infrastructure by developing clear standards for certification, interoperability, ethical guidelines, and long-term integration into healthcare delivery. The establishment of innovation hubs, public-private collaborations, and participatory policymaking processes can further bridge the gap between national strategies and grassroots health needs.

Another challenge concerns the overreliance on generalized adoption models, such as TAM and UTAUT, which are often applied across various types of health without adjustment for domain-specific needs [23], [39], [60], [66]. While these models provide valuable general insight, they do not sufficiently capture the unique behavioral, clinical, and contextual factors tied to areas such as maternal health, mental health, chronic disease management, or elderly care. Future research must prioritize the development of domain-specific adoption frameworks that incorporate relevant health constructs such as perceived disease threat, mental health or stigma. These tailored models would enhance both theoretical precision and practical relevance.

Methodological gaps also persist, with research dominated by quantitative approaches such as surveys and structural equation modeling [14], [27], [52], [53], [54], [65]. While these methods offer statistical rigor and generalizability, they often fail to capture lived experiences, cultural nuances, and emotional responses that shape real-world adoption. Qualitative methods such as ethnography, longitudinal case studies, or diary methods remain underutilized but could provide critical depth. Mixed-methods research that combines quantitative generalizability with qualitative richness would deliver a more holistic understanding of health application adoption.

In addition, cultural and socio-economic factors remain underexplored. Health applications are embedded within broader social contexts shaped by traditions, norms, and inequalities. Income level, job security, basic smartphone access or even mobile data directly affect adoption capacity, yet these factors are often overlooked in existing models. Future research should adopt culturally responsive and socio-economically inclusive approaches, engaging community-based in the design and testing process to ensure solutions are globally adoptable yet locally relevant.

Finally, psychological factors remain one of the most underexplored dimensions but most influential areas in digital health adoption [20], [26], [69], [72]. Elements such as health motivation, perceived vulnerability, technology-related anxiety, trust, and emotion play a significant role in whether users are willing to initiate and maintain use of health applications, particularly in sensitive health domains such as maternal care or

mental health. Yet these psychological dimensions are rarely integrated into mainstream models. Interdisciplinary collaboration with behavioral psychology and health science can enrich adoption frameworks by incorporating validated psychological constructs, thus yielding more human-centered and emotionally appropriate digital health solutions.

In summary, this review underscores key gaps in health application adoption research, including demographic bias, limited cross-context validation, and the underrepresentation of psychological, cultural, and infrastructural factors. While existing models offer strong foundations, future studies must adopt more inclusive, contextual, and interdisciplinary approaches to ensure broader relevance and impact. Complete references for each challenge and research priority are listed in Table VII.

TABLE VII CHALLENGES AND OPPORTUNITIES

Challenge	Research Gap Identified	Suggested Direction	References
Demographic bias	Most health application users are young / urban users.	Exploring rural, elderly, disabled population, or specific user segmentation (e.g., Gen Z, Gen Y, and female or male) could be considered for future research.	[12], [14], [18], [20], [26], [27], [30], [32], [34], [35], [36], [37], [38], [41], [45], [46], [51], [54], [59], [60], [62], [64], [66], [68], [70], [72], [73], [80], [85], [91]
Health & IT Literation	The user's difficulty in using m-health devices due to low health literacy or education suggests that they are not prepared to utilize m-health applications.	Design user education and support features. In order for the application to continue to be used, issues related to design and ease of use must be prioritized.	[2], [34], [44], [65], [71]
Area / cross-sectional or country	Current research model lacks cross-country validation, limiting the model's generalizability or small sampling area.	The research model used should be tested in different populations in other countries or a wide area.	[14], [20], [26], [27], [32], [38], [41], [51], [52], [53], [61], [62], [64], [77], [84]
Cost and infrastructure technology	The consideration of health applications requires a significant amount of operational development, maintenance costs, and low technology access.	Evaluate cost-efficiency, align with infrastructure readiness	[2], [44], [50], [51], [65], [70]
Security	Despite the increasing use of health applications, limited research has examined the role of security	Ensuring the security of health information systems is crucial in influencing user adoption and continued use	[34], [62], [71], [73]
Policy and government	Lack of policy integration	The aim of this study is to identify the elements within the policy readiness and regulatory dimension that may impact the successful implementation of m-health initiatives.	[2], [17], [34], [50], [64], [65], [70], [75]
Model generalization	Most of the applications evaluated were general health applications.	it is necessary to build more specific applications. (e.g., maternal apps, mental health app, disease prevention apps)	[23], [39], [60], [66]
Mix method	Most existing studies on health applications use quantitative approach.	A mixed method is needed to capture to measurable pattern and rich contextual insight.	[14], [27], [52], [53], [54], [65]
Culture and socio-economic gaps	The factor of socio-culture must be considered in the adoption of healthcare applications, as each location, gender, habit and economic status has a unique culture.	Develop culturally contextualized adoption models.	[23], [27], [28], [36], [39], [46], [61], [66], [68], [70], [74], [85]
Psychologist	Current studies on health application adoption have yet to extensively explore psychological factors	The psychological factors could serve as a potential area of future research in health app adoption studies, as these factors can influence user behavior in using health applications, such as motivation, risk perception, anxiety, and trust.	[20], [26], [69], [72]

VI. CONCLUSION

This literature review investigated trends, theories, and models used in health application adoption research from 2016 to 2025. After applying clear inclusion and exclusion criteria, 79 studies were selected and analyzed. Using the PRISMA framework, this review synthesized empirical findings to answer key research question related to adoption models, influencing factors, and future directions. The Unified Theory of Acceptance and Use of Technology (UTAUT) and Technology Acceptance Model (TAM) emerged as the most frequently employed frameworks, cited in over 60 % of the reviewed studies. However, many researchers have extended these core models by integrating contextual and behavioral constructs such as service quality, health consciousness, and personal innovativeness. These enhancements reflect a growing need to

adapt traditional models to complexities of health technology usage in diverse user environments. Despite an increase in scholarly attention over the last decade, several challenges remain. These include the lack of research involving underrepresented populations, limited cross-regional and cross-cultural validation, underutilization of qualitative and mixed-method approaches, and insufficient emphasis on infrastructural, regulator, and psychological aspects. Overall, this research still presents opportunities and challenges that can be explored and studied in the future. As illustrated in Table VII, these include aspects such as the sample respondents, research location, methods employed, more specific applications, government roles and policies, infrastructure supporting the adoption of applications, and public readiness from socioeconomic perspectives.

ACKNOWLEDGMENT

Funding for this work has been supported by Indonesian Education Scholarship (BPI), Center for Higher Education Funding and Assessment, Ministry of Higher Education, Science, and Technology of Republic Indonesia (PUSLAPDIK DIKBUD) and Endowment Funds for Education (LPDP) Number 02415 / BPPT / BPI.06 / 9 / 2023.

REFERENCES

[1] P. V. AshaRani et al., "Readiness and acceptance of ehealth services for diabetes care in the general population: Cross-sectional study," J Med Internet Res, vol. 23, no. 9, 2021, doi: 10.2196/26881.

[2] M. Samadbeik, N. Aslani, M. Maleki, and A. Garavand, "Acceptance of mobile health in medical sciences students: Applying technology acceptance model," Inform Med Unlocked, vol. 40, no. June, p. 101290, 2023, doi: 10.1016/j.imu.2023.101290.

[3] C. Jacob, E. Sezgin, A. Sanchez-Vazquez, and C. Ivory, "Sociotechnical Factors Affecting Patients' Adoption of Mobile Health Tools: Systematic Literature Review and Narrative Synthesis," JMIR Mhealth Uhealth, vol. 10, no. 5, pp. 1–27, 2022, doi: 10.2196/36284.

[4] N. Aljohani and D. Chandran, "The Adoption of Mobile Health Applications by Patients in Developing Countries: A Systematic Review," International Journal of Advanced Computer Science and Applications, vol. 12, no. 4, pp. 16–21, 2021, doi: 10.14569/IJACSA.2021.0120403.

[5] T. Ghiwaa, I. Khan, M. White, and N. Beloff, "Telemedicine Adoption for Healthcare Delivery: A Systematic Review," International Journal of Advanced Computer Science and Applications, vol. 14, no. 11, pp. 1230–1243, 2023, doi: 10.14569/IJACSA.2023.01411125.

[6] T. Ikwunne, L. Hederman, and P. J. Wall, "Design Processes for User Engagement with Mobile Health: A Systematic Review," International Journal of Advanced Computer Science and Applications, vol. 13, no. 2, pp. 291–303, 2022, doi: 10.14569/IJACSA.2022.0130235.

[7] M. Al-rawashdeh, and A. Z. 3, 2 , Pantea Keikhosrokiani 1,* , Bahari Belaton 1 , Moatsum Alawida 1, and 1, "IoT Adoption and Application for Smart Healthcare: A Systematic Review," Sensors, vol. 27, no. 4, pp. 225–237, 2022, doi: 10.5912/jcb1330.

[8] B. Kitchenham, "Guidelines for performing systematic literature reviews in software engineering," Technical report, Ver. 2.3 EBSE Technical Report. EBSE, no. October, 2007.

[9] D. Moher, A. Liberati, J. Tetzlaff, and D. G. Altman, "Preferred reporting items for systematic reviews and meta-analyses: the PRISMA statement," J Clin Epidemiol, vol. 62, no. 10, pp. 1006–1012, 2009, doi: 10.1016/j.jclinepi.2009.06.005.

[10] A. Liberati et al., The PRISMA statement for reporting systematic reviews and meta-analyses of studies that evaluate health care interventions: explanation and elaboration, vol. 62, no. 10. 2009. doi: 10.1016/j.jclinepi.2009.06.006.

[11] E. Kim and S. Han, "Determinants of continuance intention to use health apps among users over 60: A test of social cognitive model," Int J Environ Res Public Health, vol. 18, no. 19, 2021, doi: 10.3390/ijerph181910367.

[12] J. Cao, K. Kurata, Y. Lim, S. Sengoku, and K. Kodama, "Social Acceptance of Mobile Health among Young Adults in Japan: An Extension of the UTAUT Model," Int J Environ Res Public Health, vol. 19, no. 22, 2022, doi: 10.3390/ijerph192215156.

[13] N. S. Alharbi, A. S. Alghanmi, and M. Fahlevi, "Adoption of Health Mobile Apps during the COVID-19 Lockdown: A Health Belief Model Approach," Int J Environ Res Public Health, vol. 19, no. 7, 2022, doi: 10.3390/ijerph19074179.

[14] A. Alsyouf et al., "The Use of a Technology Acceptance Model (TAM) to Predict Patients' Usage of a Personal Health Record System: The Role of Security, Privacy, and Usability," Int J Environ Res Public Health, vol. 20, no. 2, 2023, doi: 10.3390/ijerph20021347.

[15] W. Liu, X. Fan, R. Ji, and Y. Jiang, "Perceived community support, users' interactions, and value co-creation in online health community: The moderating effect of social exclusion," Int J Environ Res Public Health, vol. 17, no. 1, 2020, doi: 10.3390/ijerph17010204.

[16] H. Wang, J. Zhang, Y. Luximon, M. Qin, P. Geng, and D. Tao, "The Determinants of User Acceptance of Mobile Medical Platforms: An Investigation Integrating the TPB, TAM, and Patient-Centered Factors," Int J Environ Res Public Health, vol. 19, no. 17, 2022, doi: 10.3390/ijerph191710758.

[17] P. W. Handayani, R. Indriani, and A. A. Pinem, "Mobile health readiness factors: From the perspectives of mobile health users in Indonesia," Inform Med Unlocked, vol. 24, no. February, p. 100590, 2021, doi: 10.1016/j.imu.2021.100590.

[18] A. D. Walle et al., "Intention to use wearable health devices and its predictors among diabetes mellitus patients in Amhara region referral hospitals, Ethiopia: Using modified UTAUT-2 model," Inform Med Unlocked, vol. 36, no. November 2022, p. 101157, 2023, doi: 10.1016/j.imu.2022.101157.

[19] O. Enaizan, B. Eneizan, M. Almaaitah, A. T. Al-Radaideh, and A. M. Saleh, "Effects of privacy and security on the acceptance and usage of EMR: The mediating role of trust on the basis of multiple perspectives," Inform Med Unlocked, vol. 21, 2020, doi: 10.1016/j.imu.2020.100450.

[20] D. Dhagarra, M. Goswami, and G. Kumar, "Impact of Trust and Privacy Concerns on Technology Acceptance in Healthcare: An Indian Perspective," Int J Med Inform, vol. 141, no. April, p. 104164, 2020, doi: 10.1016/j.ijmedinf.2020.104164.

[21] P. W. Handayani, A. N. Hidayanto, A. A. Pinem, I. C. Hapsari, P. I. Sandhyaduhita, and I. Budi, "Acceptance model of a Hospital Information System," Int J Med Inform, vol. 99, no. 1171, pp. 11–28, 2017, doi: 10.1016/j.ijmedinf.2016.12.004.

[22] R. Hoque and G. Sorwar, "Understanding factors influencing the adoption of mHealth by the elderly: An extension of the UTAUT model," Int J Med Inform, vol. 101, no. September 2015, pp. 75–84, 2017, doi: 10.1016/j.ijmedinf.2017.02.002.

[23] E.-M. Schomakers, C. Lidynia, L. S. Vervier, A. Calero Valdez, and M. Ziefle, "Applying an Extended UTAUT2 Model to Explain User Acceptance of Lifestyle and Therapy Mobile Health Apps: Survey Study," JMIR Mhealth Uhealth, vol. 10, no. 1, p. e27095, 2022, doi: 10.2196/27095.

[24] P. R. Palos-Sanchez, J. R. Saura, M. Á. R. Martin, and M. Aguayo-Camacho, "Toward a better understanding of the intention to use mhealth apps: Exploratory study," JMIR Mhealth Uhealth, vol. 9, no. 9, pp. 1–19, 2021, doi: 10.2196/27021.

[25] Z. Deng, Z. Hong, C. Ren, W. Zhang, and F. Xiang, "What predicts patients' adoption intention toward mhealth services in China: Empirical study," JMIR Mhealth Uhealth, vol. 6, no. 8, pp. 1–14, 2018, doi: 10.2196/mhealth.9316.

[26] M. Rajak and K. Shaw, "An extension of technology acceptance model for mHealth user adoption," Technol Soc, vol. 67, no. November, p. 101800, 2021, doi: 10.1016/j.techsoc.2021.101800.

[27] M. Z. Alam, W. Hu, M. A. Kaium, M. R. Hoque, and M. M. D. Alam, "Understanding the determinants of mHealth apps adoption in Bangladesh: A SEM-Neural network approach," Technol Soc, vol. 61, no. October 2019, p. 101255, 2020, doi: 10.1016/j.techsoc.2020.101255.

[28] S. A. Kamal, M. Shafiq, and P. Kakria, "Investigating acceptance of telemedicine services through an extended technology acceptance model (TAM)," Technol Soc, vol. 60, no. March 2019, p. 101212, 2020, doi: 10.1016/j.techsoc.2019.101212.

[29] M. R. Hoque, Y. Bao, and G. Sorwar, "Investigating factors influencing the adoption of e-Health in developing countries: A patient's perspective," Inform Health Soc Care, vol. 42, no. 1, pp. 1–17, 2017, doi: 10.3109/17538157.2015.1075541.

[30] X. Zhang, X. Han, Y. Dang, F. Meng, X. Guo, and J. Lin, "User acceptance of mobile health services from users' perspectives: The role of self-efficacy and response-efficacy in technology acceptance," Inform Health Soc Care, vol. 42, no. 2, pp. 194–206, 2017, doi: 10.1080/17538157.2016.1200053.

[31] C. Y. Lee, "An investigation of purchase behavior on spillover-effect health insurance: using an extended TAM," Int J Pharm Healthc Mark, vol. 17, no. 1, pp. 38–54, 2023, doi: 10.1108/IJPHM-10-2020-0094.

[32] G. Aydin, "Increasing mobile health application usage among Generation Z members: evidence from the UTAUT model," Int J Pharm Healthc Mark, vol. 17, no. 3, pp. 353–379, 2023, doi: 10.1108/IJPHM-02-2021-0030.

[33] M. A. Kaium, Y. Bao, M. Z. Alam, and M. R. Hoque, "Understanding continuance usage intention of mHealth in a developing country: An empirical investigation," Int J Pharm Healthc Mark, vol. 14, no. 2, pp. 251–272, 2020, doi: 10.1108/IJPHM-06-2019-0041.

[34] P. E. Idoga, M. Toycan, H. Nadiri, and E. Çelebi, "Factors Affecting the Successful Adoption of e-Health Cloud Based Health System from Healthcare Consumers' Perspective," IEEE Access, vol. 6, pp. 71216–71228, 2018, doi: 10.1109/ACCESS.2018.2881489.

[35] A. M. E. Elkhalifa, B. H. Ahmed, and W. M. Elmobarak, "Factors That Influence E-Health Applications From Patients' Perspective in the Kingdom of Saudi Arabia: An Exploratory Study," IEEE Access, vol. 10, no. October, pp. 109029–109042, 2022, doi: 10.1109/ACCESS.2022.3214203.

[36] S. Neeragatti, R. K. Dehury, and N. Sripathi, "Determinants of Digital Health Information Search (Dhis) Behaviour: Extending Utaut With Healthcare Behaviour Constructs," Asia Pacific Journal of Health Management, vol. 18, no. 1, pp. 1–11, 2023, doi: 10.24083/apjhm.v18i1.1685.

[37] W. Chiu and H. Cho, "The role of technology readiness in individuals' intention to use health and fitness applications: a comparison between users and non-users," Asia Pacific Journal of Marketing and Logistics, vol. 33, no. 3, pp. 807–825, 2021, doi: 10.1108/APJML-09-2019-0534.

[38] A. Ahmad, T. Rasul, A. Yousaf, and U. Zaman, "Understanding factors influencing elderly diabetic patients' continuance intention to use digital health wearables: Extending the technology acceptance model (TAM)," Journal of Open Innovation: Technology, Market, and Complexity, vol. 6, no. 3, p. 81, 2020, doi: 10.3390/JOITMC6030081.

[39] G. S. Octavius and F. Antonio, "Antecedents of Intention to Adopt Mobile Health (mHealth) Application and Its Impact on Intention to Recommend: An Evidence from Indonesian Customers," Int J Telemed Appl, vol. 2021, no. March 2019, 2021, doi: 10.1155/2021/6698627.

[40] A. Alhashem et al., "Adoption of antenatal care conversation mapping among health care providers in Saudi Arabia: Application of the diffusion innovation theory," PLoS One, vol. 18, no. 6 June, pp. 1–14, 2023, doi: 10.1371/journal.pone.0286656.

[41] J. Tavares and T. Oliveira, "New integrated model approach to understand the factors that drive electronic health record portal adoption: Cross-sectional national survey," J Med Internet Res, vol. 20, no. 11, pp. 1–17, 2018, doi: 10.2196/11032.

[42] Q. Yang, A. Al Mamun, N. Hayat, M. F. Mohd, A. A. Salameh, and Z. K. M. Makhbul, "Predicting the Mass Adoption of eDoctor Apps During COVID-19 in China Using Hybrid SEM-Neural Network Analysis," Front Public Health, vol. 10, no. April, pp. 1–17, 2022, doi: 10.3389/fpubh.2022.889410.

[43] A. Alaiad, M. Alsharo, and Y. Alnsour, "The Determinants of M-Health Adoption in Developing Countries: An Empirical Investigation," Appl Clin Inform, vol. 10, no. 5, pp. 820–840, 2019, doi: 10.1055/s-0039-1697906.

[44] H. K. Namatovu, T. J. Oyana, and H. G. Sol, "Barriers to eHealth adoption in routine antenatal care practices: Perspectives of expectant mothers in Uganda – A qualitative study using the unified theory of acceptance and use of technology model," Digit Health, vol. 7, 2021, doi: 10.1177/20552076211064406.

[45] T. Khan, K. D. Khan, M. S. Azhar, S. N. A. Shah, M. M. Uddin, and T. H. Khan, "Mobile health services and the elderly: Assessing the determinants of technology adoption readiness in Pakistan," J Public Aff, vol. 22, no. 4, 2022, doi: 10.1002/pa.2685.

[46] M. Z. Alam and L. Khanam, "Comparison of the young aged and elderly female users' adoption of mHealth services," Health Care Women Int, vol. 43, no. 10–11, pp. 1259–1283, 2022, doi: 10.1080/07399332.2022.2039149.

[47] Z. Zhu, Y. Liu, X. Che, and X. Chen, "Moderating factors influencing adoption of a mobile chronic disease management system in China," Inform Health Soc Care, vol. 43, no. 1, pp. 22–41, 2018, doi: 10.1080/17538157.2016.1255631.

[48] R. Miao et al., "Factors that influence users' adoption intention of mobile health: a structural equation modeling approach," Int J Prod Res, vol. 55, no. 19, pp. 5801–5815, 2017, doi: 10.1080/00207543.2017.1336681.

[49] J. Apolinário-Hagen, M. Menzel, S. Hennemann, and C. Salewski, "Acceptance of mobile health apps for disease management among people with multiple sclerosis: web-based survey study," JMIR Form Res, vol. 2, no. 2, pp. 1–18, 2018, doi: 10.2196/11977.

[50] H. S. W. Nugroho, H. B. Notobroto, and L. Rosyanti, "Acceptance model of a mandatory health information system in Indonesia," Healthc Inform Res, vol. 27, no. 2, pp. 127–136, 2021, doi: 10.4258/HIR.2021.27.2.127.

[51] N. Hasan and Y. Bao, A mixed-method approach to assess users' intention to use mobile health (mHealth) using PLS-SEM and fsQCA, vol. 74, no. 4. 2022. doi: 10.1108/AJIM-07-2021-0211.

[52] N. Mathai, T. McGill, and D. Toohey, "Factors Influencing Consumer Adoption of Electronic Health Records," Journal of Computer Information Systems, vol. 62, no. 2, pp. 267–277, 2022, doi: 10.1080/08874417.2020.1802788.

[53] D. Gu et al., "Assessing the Adoption of e-Health Technology in a Developing Country: An Extension of the UTAUT Model," Sage Open, vol. 11, no. 3, 2021, doi: 10.1177/21582440211027565.

[54] Y. Liu, X. Lu, G. Zhao, C. Li, and J. Shi, "Adoption of mobile health services using the unified theory of acceptance and use of technology model: Self-efficacy and privacy concerns," Front Psychol, vol. 13, no. August, pp. 1–20, 2022, doi: 10.3389/fpsyg.2022.944976.

[55] I. B. Hassan, M. A. A. Murad, I. El-Shekeil, and J. Liu, "Extending the UTAUT2 Model with a Privacy Calculus Model to Enhance the Adoption of a Health Information Application in Malaysia," Informatics, vol. 9, no. 2, 2022, doi: 10.3390/informatics9020031.

[56] G. R. El Said, "Factors Affecting mHealth Technology Adoption in Developing Countries: The Case of Egypt," Computers, vol. 12, no. 1, p. 9, 2022, doi: 10.3390/computers12010009.

[57] H. Alquran, S. Banitaan, T. Bari, Y. Chavarkar, and A. Bellamy, "The impact of trust, comfortability, usability and technophobia factors on acceptance of health information technology," Telematics and Informatics Reports, vol. 15, no. July, p. 100159, 2024, doi: 10.1016/j.teler.2024.100159.

[58] H. Al Halbusi, K. Al-Sulaiti, F. Abdelfattah, A. B. Ahmad, and S. Hassan, "Understanding consumers' adoption of e-pharmacy in Qatar: applying the unified theory of acceptance and use of technology," Journal of Science and Technology Policy Management, vol. 16, no. 3, pp. 479–505, 2024, doi: 10.1108/JSTPM-03-2023-0042.

[59] P. Stehr, L. Ermel, C. Rossmann, D. Reifegerste, A. K. Lindemann, and A. Schulze, "A Mobile Health Information Behavior Model: Theoretical Development and Mixed-Method Testing in the Context of Mobile Apps on Child Poisoning Prevention," J Health Commun, vol. 28, no. 10, pp. 648–657, 2023, doi: 10.1080/10810730.2023.2250313.

[60] M. Yan, R. Filieri, E. Raguseo, and M. Gorton, "Mobile apps for healthy living: Factors influencing continuance intention for health apps," Technol Forecast Soc Change, vol. 166, no. February, p. 120644, 2021, doi: 10.1016/j.techfore.2021.120644.

[61] Z. Barua and A. Barua, "Modeling the predictors of mobile health adoption by Rohingya Refugees in Bangladesh: An extension of

UTAUT2 using combined SEM-Neural network approach," J Migr Health, vol. 8, no. May 2021, p. 100201, 2023, doi: 10.1016/j.jmh.2023.100201.

[62] M. M. D. Alam, M. Z. Alam, S. A. Rahman, and S. K. Taghizadeh, "Factors influencing mHealth adoption and its impact on mental well-being during COVID-19 pandemic: A SEM-ANN approach," J Biomed Inform, vol. 116, no. February, p. 103722, 2021, doi: 10.1016/j.jbi.2021.103722.

[63] X. Zhang, S. Liu, L. Wang, Y. Zhang, and J. Wang, "Mobile health service adoption in China: Integration of theory of planned behavior, protection motivation theory and personal health differences," Online Information Review, vol. 44, no. 1, pp. 1–23, 2020, doi: 10.1108/OIR-11-2016-0339.

[64] S. Rana, U. Tandon, and H. Kumar, "Understanding medical service quality, system quality and information quality of Tele-Health for sustainable development in the Indian context," Kybernetes, 2023, doi: 10.1108/K-01-2023-0005.

[65] Y. Du et al., "Factors Influencing Adoption and Use of Telemedicine Services in Rural Areas of China: Mixed Methods Study," JMIR Public Health Surveill, vol. 8, no. 12, 2022, doi: 10.2196/40771.

[66] D. Tao, Z. Chen, M. Qin, and M. Cheng, "Modeling Consumer Acceptance and Usage Behaviors of m-Health: An Integrated Model of Self-Determination Theory, Task–Technology Fit, and the Technology Acceptance Model," Healthcare (Switzerland), vol. 11, no. 11, 2023, doi: 10.3390/healthcare11111550.

[67] C. Wu and G. G. Lim, "Investigating older adults users' willingness to adopt wearable devices by integrating the technology acceptance model (UTAUT2) and the Technology Readiness Index theory," Front Public Health, vol. 12, no. September, p. 1449594, 2024, doi: 10.3389/fpubh.2024.1449594.

[68] M. Z. Alam, M. R. Hoque, W. Hu, and Z. Barua, "Factors influencing the adoption of mHealth services in a developing country: A patient-centric study," Int J Inf Manage, vol. 50, no. October 2018, pp. 128–143, 2020, doi: 10.1016/j.ijinfomgt.2019.04.016.

[69] N. Hossain, F. Yokota, N. Sultana, and A. Ahmed, "Factors Influencing Rural End-Users' Acceptance of e-Health in Developing Countries: A study on Portable Health Clinic in Bangladesh," Telemedicine and e-Health, vol. 25, no. 3, pp. 221–229, 2019, doi: 10.1089/tmj.2018.0039.

[70] J. U. Palas, G. Sorwar, M. R. Hoque, and A. Sivabalan, "Factors influencing the elderly's adoption of mHealth: an empirical study using extended UTAUT2 model," BMC Med Inform Decis Mak, vol. 22, no. 1, pp. 1–21, 2022, doi: 10.1186/s12911-022-01917-3.

[71] D. Riana, A. N. Hidayanto, S. Hadianti, and D. Napitupulu, "Integrative factors of e-health laboratory adoption: A case of Indonesia," Future Internet, vol. 13, no. 2, pp. 1–27, 2021, doi: 10.3390/fi13020026.

[72] S. Alexandra, P. W. Handayani, and F. Azzahro, "Indonesian hospital telemedicine acceptance model: the influence of user behavior and technological dimensions," Heliyon, vol. 7, no. 12, 2021, doi: 10.1016/j.heliyon.2021.e08599.

[73] B. Chow and N. Legowo, "Factors Influencing User Satisfaction of PeduliLindungi App with UTAUT & Delone Mclean Models: A Case Study in Indonesia," Journal of System and Management Sciences, vol. 13, no. 2, pp. 170–184, 2023, doi: 10.33168/JSMS.2023.0212.

[74] V. Pujani, H. Dasman, R. Semiaty, and R. Nazir, "Readiness Model in Adopting E-health: An Indonesian Experience," TEM Journal, vol. 10, no. 1, pp. 95–104, 2021, doi: 10.18421/TEM101-12.

[75] A. M. Albar and M. R. Hoque, "Patient Acceptance of e-Health Services in Saudi Arabia: An Integrative Perspective," Telemedicine and e-Health, vol. 25, no. 9, pp. 847–852, 2019, doi: 10.1089/tmj.2018.0107.

[76] N. Aljohani, "Digital Health Transformation in Saudi Arabia: Examining the Impact of Health Information Seeking on M-Health Adoption during the COVID-19 Pandemic," Engineering, Technology and Applied Science Research, vol. 15, no. 1, pp. 19933–19940, 2025, doi: 10.48084/etasr.8747.

[77] A. Dash and S. K. Mohanty, "Technology readiness and the older citizen's acceptance of m-health services in India," Digital Policy, Regulation and Governance , vol. 25, no. 2, pp. 169–183, Mar. 2023, doi: 10.1108/DPRG-11-2022-0126/FULL/PDF.

[78] Y. S. 1 Avishek Choudhury 1,* and M. M. C. 3 1 and Ashish D. Nimbarte 1 , Krishnendu Sarkar 2, "Exploring Perceptions and Needs of Mobile Health Interventions for Nutrition, Anemia, and Preeclampsia among Pregnant Women in Underprivileged Indian Communities: A Cross-Sectional Survey," Nutrients, pp. 1–18, 2023.

[79] S. Rahi, M. M. Khan, and M. Alghizzawi, "Factors influencing the adoption of telemedicine health services during COVID-19 pandemic crisis: an integrative research model," Enterp Inf Syst, vol. 15, no. 6, pp. 769–793, 2021, doi: 10.1080/17517575.2020.1850872.

[80] P. Duarte and J. C. Pinho, "A mixed methods UTAUT2-based approach to assess mobile health adoption," J Bus Res, vol. 102, no. May, pp. 140–150, 2019, doi: 10.1016/j.jbusres.2019.05.022.

[81] O. Enaizan, B. Eneizan, M. Almaaitah, A. T. Al-Radaideh, and A. M. Saleh, "Effects of privacy and security on the acceptance and usage of EMR: The mediating role of trust on the basis of multiple perspectives," Inform Med Unlocked, vol. 21, p. 100450, 2020, doi: 10.1016/j.imu.2020.100450.

[82] I. B. Hassan, M. Azrifah, A. Murad, I. El-shekeil, and J. Liu, "Extending the UTAUT2 Model with a Privacy Calculus Model," Informatics (MDPI), vol. 9, no. 31, 2022.

[83] A. Garavand, M. Samadbeik, H. Nadri, B. Rahimi, and H. Asadi, "Effective Factors in Adoption of Mobile Health Applications between Medical Sciences Students Using the UTAUT Model," Methods Inf Med, vol. 58, no. 4–5, pp. 131–139, 2019, doi: 10.1055/s-0040-1701607.

[84] L. Leung and C. Chen, "E-health/m-health adoption and lifestyle improvements: Exploring the roles of technology readiness, the expectation-confirmation model, and health-related information activities," Telecomm Policy, vol. 43, no. 6, pp. 563–575, 2019, doi: 10.1016/j.telpol.2019.01.005.

[85] S. Yusif, A. Hafeez-Baig, and J. Soar, "A model for evaluating eHealth preparedness – a case study approach," Transforming Government: People, Process and Policy, vol. 14, no. 3, pp. 561–587, 2020, doi: 10.1108/TG-07-2019-0069.

[86] P. Chaodeethirathkul and S. Pankham, "Developing a Model of Actual Adoption Technology and Innovation for Health in Thailand," Academic Journal of Interdisciplinary Studies, vol. 12, no. 5, pp. 168–181, 2023, doi: 10.36941/ajis-2023-0135.

[87] M. A. Saare, A. Hussain, and W. S. Yue, "Conceptualizing mobile health application use intention and adoption among Iraqian older adults: From the perspective of expanded technology acceptance model," International Journal of Interactive Mobile Technologies, vol. 13, no. 10, pp. 28–41, 2019, doi: 10.3991/ijim.v13i10.11285.

[88] L. Schuster, V. Tossan, and J. Drennan, "Consumer Acceptance of mHealth Services: A Comparison of Behavioral Intention Models," Services Marketing Quarterly, vol. 38, no. 2, pp. 115–128, 2017, doi: 10.1080/15332969.2017.1289791.

[89] B. Chimbo and L. Motsi, "The Effects of Electronic Health Records on Medical Error Reduction: Extension of the DeLone and McLean Information System Success Model," JMIR Med Inform, vol. 12, pp. 1–12, 2024, doi: 10.2196/54572.

[90] U. Ikenyei and N. Haggerty, "Validating the Delone and Mclean's model in a developing country's infectious disease pandemic context.," BMC Infect Dis, vol. 24, no. 1, pp. 1–9, 2024, doi: 10.1186/s12879-024-09483-x.

[91] A. Alharbi and R. Ramirez, "Factors affecting electronic health record adoption in developing countries: A case of Saudi Arabia," in 26th Americas Conference on Information Systems, AMCIS 2020, 2020. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85097714964&partnerID=40&md5=df20f4c5fb241d2a33102e820a1eae29.

A New Method for Real-Time Fall Detection Based on MediaPipe Pose Estimation and LSTM

Puwadol Sirikongtham, Apichaya Nimkoompai
Faculty of Information Technology, Thai-Nichi Institute of Technology, Thailand

***Abstract*—Falls are a significant health problem among older adults, leading to serious injuries and adversely affecting both quality of life and public health burdens. Although various fall detection systems have been developed using technologies such as wearable sensors and image processing (computer vision), limitations remain in dimensions of convenience, accuracy, and real-time responsiveness. To overcome these limitations, this research aimed to present a real-time fall detection system that integrates MediaPipe pose estimation technology with a Long Short-Term Memory (LSTM) neural network. The proposed method functioned through two main components. MediaPipe pose estimation technology was applied to detect and track keypoints on the human body from real-time video input; meanwhile, a trained LSTM model was utilized to analyze the sequence of movements of the detected keypoints for classifying and differentiating between fall behaviors and normal activities. The system was trained, and its performance was evaluated using the standard UR Fall Detection Dataset. From experimental results, the proposed system achieved high efficiency in fall detection, with an accuracy of 95.2% on the test dataset. The integrated system had its capability to detect all actual fall events (with a recall of 100%). Its false positive rate was low. Compared to other research, the proposed method provided higher accuracy. These results indicated that the proposed system has the potential for practical application as an effective tool for real-time fall alerts, enabling timely assistance for those injured from falls.**

***Keywords*—Fall detection; older adults; real-time; MediaPipe; LSTM; computer vision; pose estimation**

I. Introduction

Falls among older adults have long been recognized as a critical global health problem, leading to severe injuries, long-term disabilities, and even fatalities. According to the World Health Organization (WHO), approximately 28 to 35% of individuals aged 65 and above experience at least one fall annually, with 20 to 30% of these cases resulting in moderate to severe injuries [1]. Beyond the medical consequences, falls impose a substantial economic and social burden on families and healthcare systems worldwide [2]. As populations continue to age, the demand for effective and efficient fall detection solutions has become increasingly urgent, both to improve the quality of life for older adults and to alleviate the strain on healthcare infrastructures. Over the past decade, researchers have developed a variety of fall detection systems utilizing different technologies, such as wearable sensors and vision-based approaches. Wearable devices, while capable of capturing physiological and motion data with reasonable accuracy, are often uncomfortable for continuous use and may be resisted by older adults [3]. Conversely, vision-based systems, particularly those employing image processing and deep learning, can provide non-intrusive monitoring but frequently encounter challenges related to real-time responsiveness, computational efficiency, and accuracy under diverse environmental conditions [4]. In addition, many vision-based approaches rely on spatial features alone, without adequately modeling the temporal dynamics of human motion, which are essential for distinguishing fall incidents from similar activities such as sitting or bending [11]. To address these limitations, this study proposes a novel real-time fall detection method that integrates MediaPipe pose estimation with a Long Short-Term Memory (LSTM) neural network. MediaPipe enables lightweight and efficient detection of human body landmarks across multiple platforms [5] [6] [7], while LSTM is well-suited for learning temporal sequences of movement patterns [8] [9]. By combining these technologies, the proposed system aims to achieve high accuracy in fall detection while maintaining practical feasibility for real-time applications. Experimental evaluation was conducted using the UR Fall Detection Dataset, a widely adopted benchmark in the field [12], to ensure the reliability and comparability of results. The contributions of this study are threefold: 1) the development of a real-time fall detection framework that effectively integrates MediaPipe and LSTM for temporal motion analysis, 2) empirical validation on a standard dataset, demonstrating superior performance with an accuracy of 95.2% and a recall of 100%.

II. Related Work

A. MediaPipe Pose Estimation

Pose estimation technology was applied as a fundamental component of this research, particularly the MediaPipe Pose Estimation developed by Google Research. This was an end-to-end framework designed for real-time human pose detection and tracking. This technology utilized an optimized BlazePose architecture, enabling the detection and continuous tracking of up to 33 key body landmarks with high efficiency, processing at a rate of up to 30 frames per second on standard hardware [5].

The MediaPipe Pose Estimation system consisted of two main components. First, the BlazePose Detector was utilized to identify the initial location of the body within an image. Second, the BlazePose Landmark Model provided detailed localization of landmarks on the body. The study by Bazarevsky [6] showed that MediaPipe achieved high accuracy in both controlled and real-world environments with an accuracy rate of up to 87.8% on the MS Coco dataset. Furthermore, the system was capable of running on common mobile devices without requiring high-end GPUs.

Key advantages of MediaPipe Pose Technology include:

1) Real-time capability through specialized architecture optimization, enabling processing speeds fast enough for immediate application.

2) High accuracy, performing well even in conditions with partial obstructions or incomplete viewpoints.

3) Multi-platform support, capable of running on both mobile devices and general-purpose computers.

4) 3D coordinate calculation, in addition to 2D image pose estimation, it can also estimate positions in a 3D world.

In the context of fall detection, MediaPipe Pose Technology provides several crucial pieces of information, namely:

- The positions of various body joints
- Angles between key joints
- The speed and movement patterns of joints

The operation of MediaPipe Pose Technology begins with receiving an input image through the following steps:

1) Person Detection
2) Landmark Prediction
3) Post-processing
4) Output Generation by exporting results in 2D image coordinates and 3D world coordinates.

Numerous studies, such as that of Chiu [7], have proved that MediaPipe Pose achieved sufficient accuracy for applications in medical motion analysis and health-related monitoring tasks, while maintaining a processing speed that is suitable for real-time systems.

B. LSTM is a Type of Recurrent Neural Network (RNN)

LSTM is designed to address the issue of long-term dependencies. This is suitable for analyzing sequential data such as video or continuous human motion [8]. In this study, LSTM techniques were applied to learn the sequence of keypoints obtained from MediaPipe in order to classify fall behaviors among older adults. The LSTM function relied on its architecture, comprising a cell state, input gate, output gate, and forget gate, which collectively enable the network to retain long-term information and mitigate the vanishing gradient problem [9]. LSTM was capable of effectively analyzing motion patterns and learn the distinctive characteristics of individual falls.

C. Integration of MediaPipe and LSTM

When the camera captures video footage, the system utilizes MediaPipe to detect and track the positions of keypoints on the human body. These keypoints are then arranged in chronological order and fed into the LSTM to analyze movement patterns and classify whether an event is a fall. The integration of MediaPipe and LSTM allows the system to operate in real-time, with high accuracy and rapid response, which is critical in emergency situations. This design also helps to reduce the likelihood of false alarms and enhances the reliability of the system in real-world applications.

D. Related Research on Fall Detection

Fall detection has been a consistently significant topic in healthcare research, particularly among older adults who are at high risk of serious health consequences from falls. Various technologies have been applied to develop fall detection systems, including wearable sensors, computer vision systems, and artificial intelligence techniques to enhance behavior classification performance.

1) Sensor detection systems: The study by Kepski and Kwolek [10] developed a system using an accelerometer and a depth sensor to distinguish fall behaviors from normal activities. Although the system proved to have high accuracy, its reliance on specialized hardware limits its practical implementations. Similarly, Kulurkar [3] proposed a fall detection system that utilizes wearable sensors integrated with IoT technology in smart homes, enabling real-time alerts. However, this technique had limitations in terms of the inconvenience of wearing it for older adults.

2) Image and video processing system: Recently, vision-based systems have gained popularity, especially those incorporating deep learning techniques. Although the use of Convolutional Neural Networks (CNN) in the work by Saleh and Jeannès [11] has shown effectiveness in detecting falls from video due to CNN's capability in extracting spatial features from still images or individual video frames, a major limitation of this approach is the lack of temporal analysis. CNN treats video as a series of still images without accounting for the sequence or the transitional dynamics of pose before and after a potential fall. This temporal context is essential for distinguishing falls from other behaviors, such as sitting down or bending to pick something up.

The absence of temporal components may lead to confusion between similar movements that have different outcomes, such as, a rapid collapse caused by a fall versus an intentional quick sit-down. Without a model capable of learning from event sequences, the system could easily produce false positives or false negatives. Therefore, it is essential to utilize a model that can capture continuous movement patterns, such as LSTM, to enhance the accuracy of the analysis and ensure better contextual alignment with the actual event.

III. METHODOLOGY

A. System Architecture

The architecture of the proposed method is an integrated system of MediaPipe with an LSTM-based classifier to monitor and evaluate the movements of older adults in real-time. For its functionality, webcam video capture was utilized to continuously record motion data. Data were then processed using computer vision technology to extract pose data. The LSTM model has been trained on a dataset involving both general movement behaviors and fall incidents. These data were analyzed through the LSTM model to accurately detect fall events. The integration of these components is a key element of the proposed solution, ensuring effective hazard prevention and promoting the safety of older adults in their daily lives. The workflow diagram of the proposed system is shown in Fig. 1.

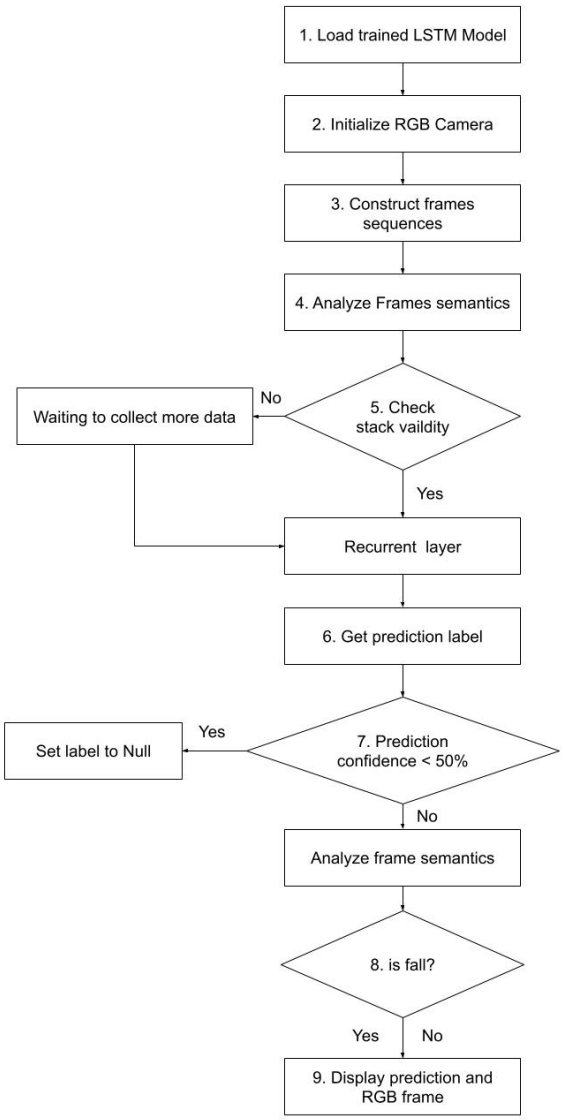


Fig. 1. Workflow diagram of the proposed real-time fall detection method using MediaPipe Pose Estimation and LSTM.

Real-Time Fall Detection System Workflow

1) Load the LSTM model along with its trained weights, which are capable of classifying movement behaviors (between normal activities and fall).

2) Initialize RGB camera by activating the webcam to capture real-time video input.

3) Generate frame sequence by extracting a sequence of consecutive video frames to gather sufficient motion data for analysis.

4) Analyze frames Using MediaPipe to extract pose landmarks of individuals in each frame.

5) Verify sequence accuracy to check whether the frame sequence is complete and sufficient.

- If incomplete, continue accumulating data.
- If complete, forward the sequence to the LSTM model for behavior classification.

6) Receive classification output from LSTM by obtaining a confidence score indicating whether the detected behavior corresponds to a "fall" or "non-fall" event.

7) Evaluate prediction confidence:

- If the confidence score is below 50%, label the prediction as null.
- If the confidence score exceeds 50%, proceed with further analysis.

8) Analyze the classification result by assessing whether the movement corresponds to normal behavior or a fall incident.

9) Display prediction results by presenting the classification result ("fall" or "non-fall") along with the relevant video frames for monitoring or alert.

B. Dataset Details

For developing this system, the dataset was derived from the UR Fall Detection Dataset [12], which consists of a total of 70 video clips. These are divided into 30 videos showing fall incidents and 40 videos capturing normal movement behaviors. The videos were recorded from various camera angles and perspectives. Participants with different heights and physical characteristics were included to reflect a broad range of real-world motion patterns.

During the data preparation process, MediaPipe was applied to extract keypoints from each video frame, as shown in Fig. 2. These keypoints were then stored in a clearly structured folder. The dataset covers sequences of various movement behaviors, including walking, sitting, and different types of falls, making it well-suited for training the LSTM model to effectively learn and detect fall events. Detailed characteristics of the dataset are presented in Table I.

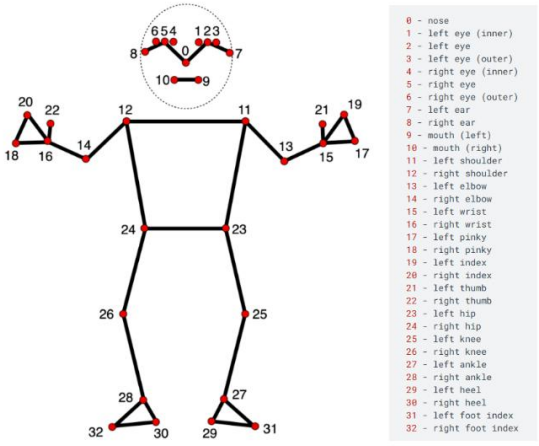


Fig. 2. Example of 33 human body landmarks identified using MediaPipe.

C. Implementation

In the initial phase of system implementation, real-time human pose detection was conducted using the MediaPipe Holistic model. Images were continuously captured via an RGB

camera and processes to extract landmark data of various body parts, such as the pose, face, and hands. These landmark data are considered key attributes for fall detection.

Long Short-Term Memory (LSTM) neural network is the system's core, which was designed to effectively learn time-sequenced data. The model incorporates multiple LSTM layers followed by Dense layers, which classify whether the user belongs to a "fall" or "non-fall" condition. This model is trained using the Adam optimizer and utilizes Categorical Cross-Entropy as the loss function.

A key feature of the LSTM layers is their ability to retain and learn behavioral patterns or motion sequences over a continuous period of time. This capability makes them well-suited for distinguishing between normal behaviors (such as walking or sitting) and abnormal events such as falling. The internal mechanism of the LSTM relies on gates that manage the flow of information within the memory units during each time step. These two components, including the landmark extraction and the LSTM-based classification, are driven by their different loss functions.

TABLE I COMPOSITION OF DATASET DETAILS

Posture Type	Falling	Normal
Number of Videos	30	40
Frames per Video	30	30
Landmarks per Frame	33	33
Total Keypoint Values Per Frame	132	132

Total Keypoint Values per Frame = Number of Landmarks × Number of Coordinate Values per Landmark Therefore, 33 Landmarks × 4 values per Landmark = 132 values per frame.

In this research, the MediaPipe Pose model utilized 3D coordinate values (x, y, z) + a confidence score, totaling four values per keypoint. These values were used as inputs for the LSTM model to learn pose characteristics. The confidence score assisted in filtering out uncertainly detected landmarks. A total of 33 keypoints (including 11 keypoints located in the head region) were used in the analysis. These points are significant to identify abnormal movement patterns, such as tilting or sudden head-dropping motions, which are often the initial signs of falls.

In this study, the following equations show the mechanism of the Long Short-Term Memory (LSTM) cells applied to analyze movement sequences extracted via MediaPipe:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t]) + b_i \quad (1)$$

Input Gate [Eq. (1)] controls the amount of new information from the current input (x_t) and the previous hidden state (h_{t-1}) that will be recorded in the cell state. A value close to 1 indicates a high allowance of information flow, whereas a value close to 0 blocks the information.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t]) + b_f \quad (2)$$

Forget Gate [Eq. (2)] determines whether to retain or discard the old information in the cell state (C_{t-1}). This eliminates irrelevant data, such as normal posture prior to a fall. A value close to 1 indicates the retained information, whereas a value close to 0 indicates the forgotten information.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t]) + b_o \quad (3)$$

Output Gate [Eq. (3)] controls which part of the cell state is passed on as the hidden state (h_t). This determines which aspects of the movement are important for fall detection.

$$C_t = f_t * C_{t-1} + i_t * \tanh(W_C \cdot [h_{t-1}, x_t] + b_C \quad (4)$$

Cell State Update [Eq. (4)] controls that irrelevant information is discarded via the Forget Gate (f_t), and new information is added via the Input Gate(i_t). The tanh function generates candidate values (ranging from –1 to 1) for updating the state, enabling the recording of movement patterns that may lead to a fall.

$$h_t = o_t * \tanh(C_t) \quad (5)$$

Hidden State [Eq. (5)] is the output of the LSTM cell that is passed to the next layer. It retains crucial posture information for prediction in the subsequent frame.

$$y = softmax(W \cdot h + b) \quad (6)$$

Classification Output [Eq. (6)] converts the final Hidden State into the probability of each class (fall / non-fall).

where,

x_t = Current input (keypoint values from MediaPipe at time t)

h_t = Hidden State (information passed between cells)

C_t = Cell State (long-term memory)

σ = Sigmoid (Squashing Function)

tanh = Hyperbolic Tangent

W = Weight Matrix learned to capture motion patterns from MediaPipe data.

b = Bias used to balance the activation of each gate. The fall detection system was trained using sequence data over 150 epochs. The Early Stopping technique was implemented to monitor the categorical accuracy. Training was automatically terminated once the accuracy stabilized, in order to prevent overfitting, which may arise from excessive learning on the training data.

After completion of the training process, the model was evaluated using a test dataset comprising 21 videos (12 normal activity videos and 9 fall incidents), accounting for 30% of the total dataset. The evaluation involved the generation of a confusion matrix and the calculation of an accuracy score to assess the model's ability to classify behaviors in videos it had not encountered during training.

For extracting pose features, the system utilized the MediaPipe Pose model, which can detect 33 keypoints on the human body. Among these, 11 keypoints located around the head were included due to their critical role in analyzing abnormal movements such as sudden tilting or dropping of the head which are often early signs of falls.

In real-world application, the system captures real-time video input through an RGB camera, processes it using MediaPipe to detect keypoints, and then feeds the sequence data into the pre-trained LSTM model to predict behavior. The results are displayed immediately on screen, both as text and audio alerts, to help caregivers promptly detect and respond to fall incidents.

IV. Results

A. Experimental Results

The experimental results of the fall detection system are presented in Table II, which shows the performance metrics of the developed model. The results indicate that the model was capable of accurately classifying behaviors, with only one misclassification out of a total of 21 samples. The test dataset consisted of 9 videos of actual falls (positive class) and 12 videos of normal activities (negative class). The system achieved an accuracy of 0.952, with most false positives caused by rapid sit-stand movements or sudden bending to pick up objects.

To further illustrate the system's performance Fig. 3 and Fig. 4 provide visual examples of the MediaPipe Pose Estimation and LSTM model's outputs. Fig. 3 depicts the successful detection and classification of a normal posture, where MediaPipe accurately tracks 33 keypoints, including critical head and body joints, and the LSTM model assigns a high-confidence "Normal Posture" label. In contrast, Fig. 4 shows the detection of a falling posture, capturing the rapid descent and tilt of keypoints.

TABLE II Training and Evaluation Indicators of the Model

Metric	Value	Comments
Epochs	150	Total training epochs on the UR Fall Detection dataset
Accuracy	0.952	Accuracy on the test dataset (including both fall and non-fall classes)
False Positives	1	1 False Alarm (Normal Activity)
True Negatives	11	Correctly identified normal activities 11 times
True Positives	9	All falls were successfully detected
False Negatives	0	No missed fall detections
Precision	0.9	Precision of the alert
Recall	1	Fall detection rate
F1-Score	0.947	Balance between Precision and Recall

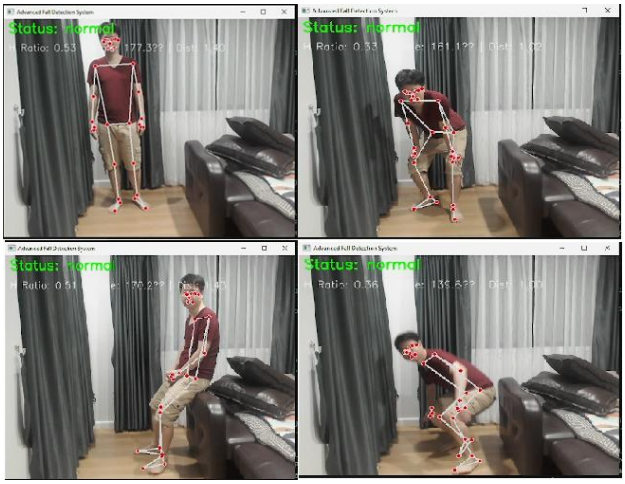


Fig. 3. Testing the MediaPipe Pose Estimation and LSTM model for normal posture.

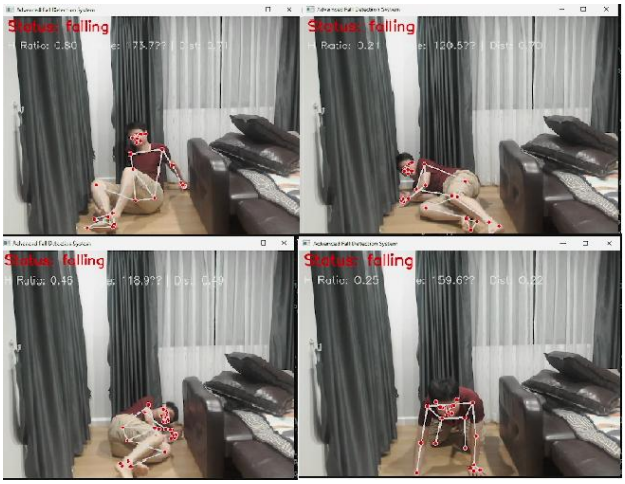


Fig. 4. Testing the MediaPipe Pose Estimation and LSTM Model for falling posture.

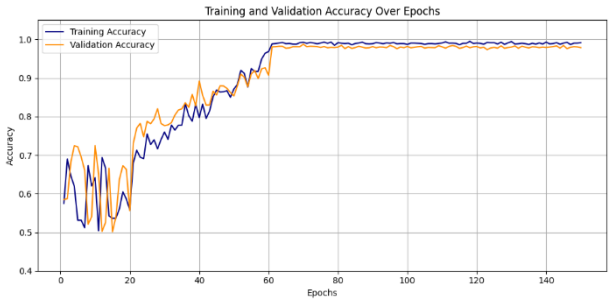


Fig. 5. Training and validation accuracy performance over epochs.

As shown in Fig. 5, the graph illustrates the model's accuracy throughout the training period, revealing a stable and steadily increasing learning trend, as the number of epochs increases.

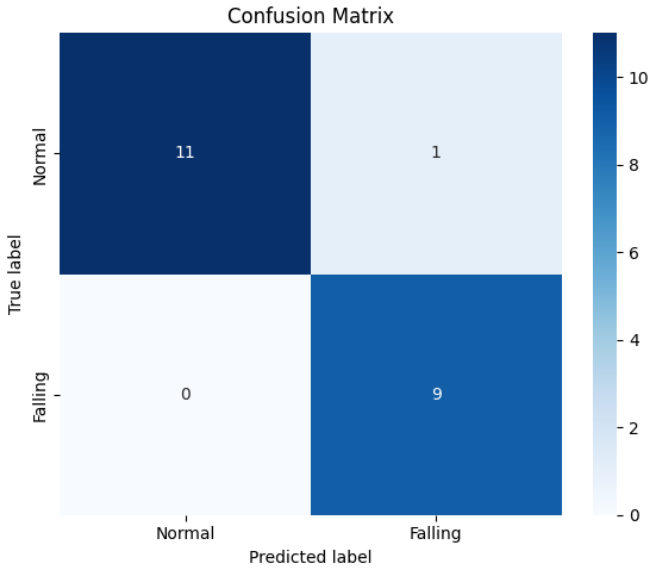


Fig. 6. Confusion matrix of the system.

Fig. 6 presents the confusion matrix, which displays the number of correctly classified instances (True Positives and True Negatives) and misclassified instances (False Positives and False Negatives). This allows for a detailed analysis of the system's error distribution. The fall detection system exhibited high performance when tested on 21 video clips, achieving a 100% recall, correctly identifying all fall incidents. The overall accuracy was 95.2%. There was only one false alarm from normal activities.

V. Discussion

A. Comparative Analysis of Pose-Based Fall Detection Models

In the development of a real-time fall detection system utilizing an LSTM neural network, comparative analysis with existing research is essential to prove the effectiveness of the proposed method. In this study, a comparative analysis was conducted with two related works. First, the research entitled "Real-Time Human Fall Detection using a Lightweight Pose Estimation Technique" [13], proposed a real-time fall detection system employing a lightweight pose estimation technique. The study utilized the Movenet model for extracting body keypoints. This system is capable of operating on resource-constrained devices without relying on cloud-based processing, thereby addressing privacy problems. The model achieved an accuracy of 91.67% on the URFD dataset [12].

Second, the research entitled "PreFallKD: Pre-Impact Fall Detection via CNN-ViT Knowledge Distillation" [14], proposed a pre-impact fall detection system by applying a knowledge distillation technique between Vision Transformer (ViT) and Convolutional Neural Network (CNN) models. This approach created a high-performance model that is also compatible with resource-constrained devices. The system achieved an accuracy of 92.66%.

TABLE III Comparative Results of different Models

Model	Latency (ms/frame)	Accuracy
MediaPipe Pose + LSTM	33 ms	0.952
MoveNet (Lightweight CNN) [13]	15 ms	0.9167
CNN + Vision Transformer (ViT) [14]	50 ms	0.9266

From Table III, the comparison of fall detection accuracy across different real-time models was presented. The proposed MediaPipe + LSTM model in this study achieved the highest accuracy at 95.2%, outperforming both previous works. The model's latency was measured at 33 ms/frame, which qualifies as real-time performance (≤50 ms), though slightly slower than the Movenet-based method.

The results of the research entitled "Real-Time Human Fall Detection using a Lightweight Pose Estimation Technique" [13] were compared, which used the Movenet model and achieved the lowest latency of only 15 ms/frame due to its single-frame structure, and an accuracy of 91.67%. It was found that despite the advantages in speed and processing on low-resource devices, the foresaid system may have limitations in learning complex event sequences or behaviors due to the lack of a temporal sequential model like an LSTM.

Meanwhile, the results of the research entitled "PreFallKD: Pre-Impact Fall Detection via CNN-ViT Knowledge Distillation" [14] achieved an accuracy of 92.66%, which was also high. However, it had the highest latency at 50 ms/frame. Its strengths lie in its pre-impact detection capability and the effective use of knowledge distillation between Vision Transformer (ViT) and CNN. Nonetheless, this model had a more complex structure and demanded more resources for training compared to the MediaPipe + LSTM system.

In summary, the MediaPipe + LSTM model offers significant advantages, including the highest accuracy among all three models (95.2%) and a latency of 33 ms/frame, which falls within the real-time threshold. It effectively learned temporal patterns of continuous poses using LSTM and efficiently extracted keypoints with MediaPipe. Furthermore, it does not require advanced hardware or cloud processing.

From these comparative experimental results, it can be proven that the proposed method was highly efficient and had significant potential for pose detection and fall risk assessment. This can be applied in nursing homes or hospitals to enhance safety and quality of life.

VI. Conclusion

In this research, an innovative real-time fall detection method was proposed utilizing human pose estimation via MediaPipe Pose in conjunction with a Long Short-Term Memory (LSTM) neural network model. The integrated technique was capable of effectively learning sequential pose data and accurately distinguished fall events. Experimental results indicated that the proposed model was functional with high accuracy in classifying both normal activities and fall incidents, exhibiting strong accuracy and a low misclassification rate.

Enhancing the diversity of training data such as varying camera angles, lighting conditions, and pose characteristics of different samples should be emphasized for future directions to improve the system's adaptability to a wider range of real-world environments. In addition, further optimization of the model should be promoted for faster performance and a more lightweight architecture. This is for the implementation with resource-constrained devices, such as standard CCTV cameras or IoT devices.

Lastly, plans should be established to improve the accuracy and responsiveness of the alert system, including connecting the system to applications for caregivers or medical personnel. This aims to enhance the effectiveness of fall prevention and care for older adults or individuals at risk. A key future goal is to evaluate the long-term effectiveness of the system through collaborations with public health agencies, in order to assess its impact on user safety and quality of life.

REFERENCES

[1] World Health Organization, "Falls," 26 April 2021. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/falls. [Accessed: May . 30, 2025].

[2] World Health Organization, *WHO Global Report on Falls Prevention in Older Age*. Geneva, Switzerland: WHO, 2007, p. 4. [Online]. Available: https://extranet.who.int/agefriendlyworld/wp-content/uploads/2014/06/WHo-Global-report-on-falls-prevention-in-older-age.pdf). [Accessed: May . 30, 2025].

[3] P. Kulurkar, C. K. Dixit, V. C. Bharathi, A. Monikavishnuvarthini, A. Dhakne, and P. Preethi, "AI-Based Elderly Fall Prediction System Using Wearable Sensors: A Smart Home-Care Technology with IoT," Measurement: Sensors, vol. 25, p. 100614, Feb. 2023, doi: 10.1016/j.measen.2022.100614.

[4] P. Sonawane, M. Patil, S. Shinde, and R. Patil, "Live Object Recognition Using YOLO," International Journal for Research in Applied Science & Engineering Technology (IJRASET), vol. 13, no. 4, pp. 1308–1311, Apr. 2025. [Online]. Available: https://www.researchgate.net/publication/380207553_Live_Object_Recognition_Using_YOLO.

[5] J. W. Kim, J. Y. Choi, E. J. Ha, and J. H. Choi, "Human Pose Estimation Using MediaPipe Pose and Optimization Method Based on a Humanoid Model," Applied Sciences, vol. 13, no. 4, Art. no. 2700, 2023. [Online]. Available: https://www.mdpi.com/2076-3417/13/4/2700.

[6] V. Bazarevsky, I. Grishchenko, K. Raveendran, F. Zhang, and M. Grundmann, "BlazePose: On-device Real-time Body Pose Tracking," arXiv preprint arXiv:2006.10204, 2020. [Online]. Available: https://arxiv.org/abs/2006.10204.

[7] Y. H. Chiu, L. C. Lin, R. C. Yang, and Y. P. Chang, "An Automatic Approach of Video-Based Landmark Detection and Movement Analysis for Assessing Symptoms of Bradykinesia in Parkinson's Disease," Journal of Medical and Biological Engineering, vol. 44, 2024. [Online]. Available: https://link.springer.com/article/10.1007/s40846-024-00876-6.

[8] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997, doi: 10.1162/neco.1997.9.8.1735.

[9] F. Ordóñez and D. Roggen, "Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition," Sensors, vol. 16, no. 1, p. 115, 2016, doi: 10.3390/s16010115.

[10] M. Kępski and B. Kwolek, "Human fall detection on embedded platform using depth maps and wireless accelerometer," Computer Methods and Programs in Biomedicine, vol. 117, no. 3, pp. 489–501, 2014, doi: 10.1016/j.cmpb.2014.09.005.

[11] B. Saleh and R. Jeannès, "Vision-Based Fall Detection Using Convolutional Neural Networks," Computers in Biology and Medicine, vol. 111, 2019, doi: 10.1016/j.compbiomed.2019.103355.

[12] M. Kępski, UR Fall Detection Dataset, Interdisciplinary Centre for Computational Modelling, University of Rzeszow. [Online]. Available: https://fenix.univ.rzeszow.pl/~mkepski/ds/uf.html.

[13] E. Alam, A. Sufian, P. Dutta, and M. Leo, "Real-Time Human Fall Detection using a Lightweight Pose Estimation Technique," arXiv preprint arXiv:2401.01587, Jan. 2024. [Online]. Available: https://arxiv.org/abs/2401.01587.

[14] T.-H. Chi, K.-C. Liu, C.-Y. Hsieh, Y. Tsao, and C.-T. Chan, "PreFallKD: Pre-Impact Fall Detection via CNN-ViT Knowledge Distillation," arXiv preprint arXiv:2303.03634, Mar. 2023. [Online]. Available: https://arxiv.org/abs/2303.03634.

Design of Marketing Digital Control System Based on the Integration of Big Data Analysis and Machine Learning

Qiming Li[1], Songling Du[2], Shiyuan Zhang[3]*
Marketing Business Department, State Grid Gansu Power Company, Lanzhou 730000, Gansu, China[1, 2]
Information Construction Division, Gansu Tongxing Intelligent Technology Development Co. Lanzhou, Lanzhou 730050, Gansu, China[3]

***Abstract*—The advent of the digital age has made it difficult for traditional marketing methods to meet the rapidly changing needs of the market. To improve the efficiency and effectiveness of enterprise marketing activities, it is particularly important to develop intelligent and precise marketing management systems. Therefore, the study proposes a marketing digital control system based on the integration of big data analysis and machine learning, which utilizes the Apache Flink distributed big data processing framework to design a marketing control system that includes system content and functional requirements. At the same time, machine learning design ideas are introduced into marketing recommendation algorithms, using reinforcement learning to enrich the business logic of the Rete network, dynamically generate and update rules, and calculate user interest while ensuring the fit between marketing recommendation content and user interest information. The study constructed a self-designed dataset (consisting of over 30000 pieces of data) through data simulation and crawling, and compared the research method with other machine learning algorithms on the same dataset. The results show that the maximum matching accuracy of the improved recommendation algorithm reaches 90.12%, and the prediction accuracy of user consumption behavior exceeds 88%, which is better than other comparative algorithms. The mean absolute error value on the product is less than 0.10, and the F1 value is greater than 0.65, indicating significant recommendation effectiveness. The research-designed marketing digital control system effectively integrates big data analysis and machine learning technology, providing support for the digital transformation of enterprises and the intelligent upgrading of related fields.**

***Keywords*—Big data; flink; reinforcement learning; rete network; interest level; term frequency-inverse document frequency; rule**

I. INTRODUCTION

With the continuous maturity and popularization of technologies such as cloud computing, big data, and artificial intelligence, enterprises can use multiple channels and methods to collect, process, and analyze user data, thereby providing reference for marketing decisions. The customer-oriented business philosophy requires enterprises to actively develop marketing activities and occupy favorable market competition shares in order to meet customer needs and respond to market dynamic changes [1]. Modern consumers are increasingly focusing on personalized and convenient shopping experiences, and their shopping needs and preferences are also complex and diverse. Strengthening digital marketing control is an important direction to achieve precise targeting of target customers and provide personalized products and services [2]. In recent years, with the emergence of diversified consumer markets, product consumption patterns have also shown a trend of diversification. Merchants with effective marketing costs often adopt a consumption rebate model to attract consumers, but the mechanism of this model is relatively single and has not fully utilized its advantages in attracting repeat customers and customer sources [3]. Data-driven decision-making has become an important part of marketing practice, and grasping target markets and user information is a key aspect of marketing management [4]. How to achieve coordinated planning of various resources and targeted product and service marketing is an important challenge facing enterprises, and how to mine information for reference and utilization in the complex and massive information data has also become a blind spot faced by enterprises. Previous research has only focused on data mining and prediction, lacking a comprehensive and detailed control design for marketing activity systems. Some of the content focuses on marketing design and often uses association algorithms, clustering algorithms, etc., for data mining, with insufficient consideration for the temporal and differential nature of marketing content data [5-6]. Therefore, the study proposes a digital marketing control system that integrates Flink real-time computing and reinforcement learning to optimize the Rete network, filling the gap mentioned above. The core goal is to achieve highly responsive and interpretable intelligent marketing decisions through dynamic rule generation and interest evolution modeling, and verify their effectiveness in complex business scenarios, thereby solving the performance bottleneck of traditional methods in complex rule sets and ensuring that marketing recommendation results are both accurate and compound with business logic. Specifically, the study proposes to combine big data analysis with machine learning algorithms to design a detailed marketing digital control system from the overall to the local level. Firstly, a processing framework is built using Apache Flink distributed big data technology. Then, reinforcement learning (RL) is used to improve the Rete network to enhance accuracy. To ensure the fit between marketing recommendation content and user interest information, user interest calculation is proposed. The innovation of the research lies in two aspects. One is to leverage Flink's advantages in

*Corresponding Author

block management and distributed parallelization, and use big data technology to achieve intelligent processing of user data. The second is to improve the Rete network by considering the performance of marketing recommendations and the increase in the number of rules, including dynamic matching, interest calculation, etc. The research aims to design a new digital marketing control scheme to help enterprises improve marketing efficiency and effectiveness, and provide support for the implementation of intelligent marketing systems.

The research mainly analyzes the relevant content of the design of the marketing digital control system from six aspects. Section II is a literature review and discussion of the data analysis and related algorithms of current marketing digital control. Section III is to design methods from two aspects: analysis of marketing digital control and construction of the Flink distributed system framework. The digital control analysis includes using a reinforcement learning network rule matching algorithm to generate rules, and considering user preferences, using a word frequency inverse document frequency algorithm to ensure the fit between marketing recommendation content and user interest information. Section IV is to test and analyze the performance comparison and application effects under the research methods. Section V and Section VI are a discussion and analysis of the results, as well as an overview summary of the entire text.

II. Related Work

Precision marketing requires a large amount of data support, including sales basic data, customer target group profile information, and product development information. With the help of big data analysis methods, enterprises can understand user needs, reduce marketing costs, and achieve low-cost, long-term, stable development, achieving precision marketing effects [7, 8]. Scholars Liu and Guo proposed using big data and fuzzy control algorithms to predict digital market preferences in the publishing industry, providing a reference for market analysis and enterprise transformation. The results indicated that this information technology could assist enterprise decision-making and organizational analysis by analyzing consumer information, and promote the development of digital publishing [9]. Rivas and Zhao scholars believed that ChatGPT has great potential for transforming marketing in the future, including understanding customers and achieving service automation [9]. Rithani et al. also believed that big data analysis can be used to collect massive amounts of data, where deep learning algorithms can achieve automatic feature extraction [10].

Liao et al. attempted to use clustering analysis and association rules to design a personalized marketing system for social networks, and believed that rule recommendations could be used to achieve personalized recommendations. The results indicate that this method can effectively achieve marketing and sales goals based on social media content [11]. Cherkaoui et al. proposed using data cleaning, variable statistics, and Apriori algorithm to achieve correlation analysis of marketing data, in order to better understand the relationship between marketing data and consumer behavior. The results indicate that this recommendation method can be combined with artificial intelligence to improve marketing efficiency [12]. Iyelolu et al. found that artificial intelligence significantly improves the accuracy of customer positioning and personalization, dynamically adjusts marketing strategies, and thus increases engagement and conversion rates [13].

Currently, the mainstream approach to precision marketing is to use big data-driven thinking, predictive models, and automated marketing tools for research. For example, reference [12] uses clustering to mine group preferences. However, most studies only focus on a single data source (such as social networks [11] or transaction logs [12]), lacking the fusion analysis of multimodal data (such as text+behavior+spatiotemporal). And traditional clustering/association rules rely on historical data, making it difficult to respond in real-time to interest drift (such as the impact of sudden heat search for products). To address these issues, innovative solutions are proposed: firstly, a multi-source real-time data processing framework based on Flink is constructed, integrating user profiles, behavior data, and external events; Secondly, design the RLHF Rete algorithm to dynamically optimize the rule network through reinforcement learning, achieving millisecond level matching; The research method can provide an interpretable and implementable technical path for intelligent marketing systems.

III. Methods

A. Design of Marketing Digital Recommendation Ideas

The workspace in most rule matching management is based on progressive rule logic, but the matching efficiency is low, and it is difficult to handle the problem of increased computational load during the periodic inspection process [14]. Therefore, the Rete algorithm is proposed for rule generation matching, which utilizes the characteristics of node sharing and state preservation to improve matching efficiency, reduce data transmission time between nodes, and display rule conditions through a data flow network, which is not affected by the number of rules and can effectively reduce the computational complexity of the product. Fig. 1 shows a schematic diagram of the Rete network.

In Fig. 1, the Rete network includes Alpha and Beta networks related to storage event testing, which respectively implement single and multiple test contents. Essentially, they can be seen as directed acyclic graphs of rule sets. When the Rete network runs, it matches rule conditions and facts through an object network in memory. Each working memory element enters the Rete network from the root node and may be stored in intermediate storage through propagation until it reaches the terminal [15]. Although the two major characteristics of Rete networks can improve operational efficiency, it should be noted that they do not consider the interrelationships between inputs, making it difficult to achieve complex connections between rule-based conditional models and factual structures. Therefore, the study utilizes Reinforcement Learning from Human Feedback (RLHF) to enrich the business logic of the Rete network, and dynamically generates and updates rules through continuous learning, namely the Reinforcement Learning from Human Feedback-Rete (RLHF-Rete) algorithm for rule matching in RL networks. The RLHF-Rete algorithm first uses RL to provide a classification model for events during rule formulation processing, allowing the system to explore in

the state space. Then, with the continuous improvement of dataset learning strategies, the rule patterns are strengthened. Fig. 2 is a schematic diagram of rule formulation processing.

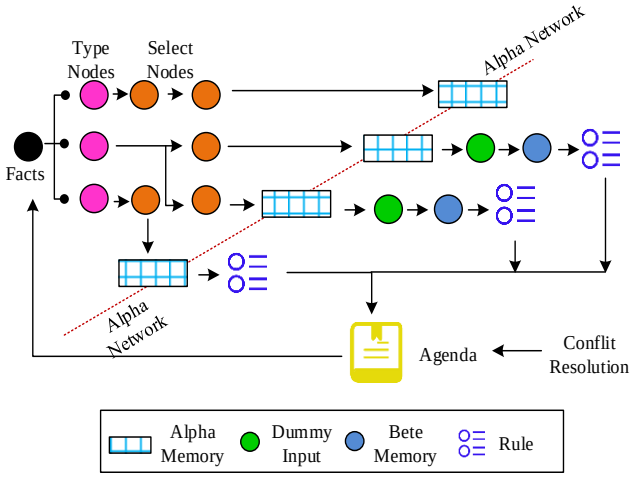


Fig. 1. Schematic diagram of Rete network.

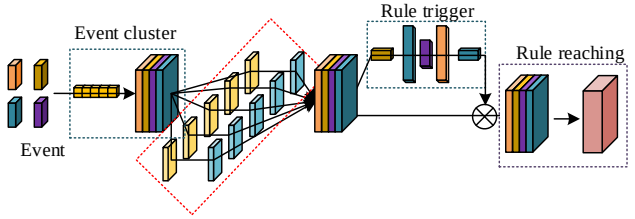


Fig. 2. Schematic diagram of rule formulation processing.

In Fig. 2, the time clusters are sorted according to a rule sequence and processed into clustering patterns. Rule triggers extract and train the event set, identify effective event patterns, and train the classification model to obtain the final rules. When RL is used for feedback control, different reward signal forms are set for different time periods based on different time point deviations, so that the controller and state will execute the new action according to the strategy. If the multi-mode rule sequence of the RLHF-Rete algorithm satisfies the Markov decision process property, it can describe the state transition probability $P(s'|s,a)$ between states and actions, as shown in Eq. (1):

$$P(s'|s,a) = Pr\{St = s'|St-1 = s, At-1 = a\} \quad (1)$$

In Eq. (1), S represents a continuous state space, A represents a finite behavior space, s represents the initial state, a represents the action, s' represents the subsequent state, r represents the reward function, and t represents time [16]. Eq. (1) can be represented as a three parameter function, and given the discount factor and strategy, the value function $V^c(s)$ can be represented as Eq. (2):

$$V^c(s) = E\left[\sum_{t=0}^{\infty} \lambda^t r(s_t, c_t(s_t))\right] \quad (2)$$

In Eq. (2), λ represents the discount factor, c is the given strategy, s_t is the behavior space, and E represents the expected value. The value range of λ generally does not exceed 1.0 to ensure that the optimal strategy is within a bounded range. At the same time, by using the likelihood ratio method to enhance the policy gradient, calculating unbiased gradient estimation, and selecting an appropriate value baseline to simplify the complexity of estimation, the updated policy gradient $\nabla_\gamma M(\gamma)$ can be expressed as Eq. (3):

$$\nabla_\gamma M(\gamma) = E_c\left[\sum_{t=0}^{T-1} \nabla_\gamma \log c(a_t|s_t,\gamma)\left(\sum_{t'=t}^{T-1} r_{t'} - v^c(s_t)\right)\right] \quad (3)$$

In Eq. (3), $M(\gamma)$ is the objective function, $r_{t'}$ is the reward, $v^c(s_t)$ is the state value function, and γ represents the parameters of $M(\gamma)$. The choice of reward function can affect the effectiveness of rule generation. Therefore, research can provide feedback on the time window, balance rule allocation by controlling the weight of reward values, and consider the impact of rule delay on decision-making, in order to achieve pattern adjustment [17]. After completing the multi rule learning reinforcement mentioned above, considering the uncertainty of the recommendation process, research is conducted on user preference design under semantic pattern extension, including user evaluation information, consumption records, user profiles, etc. This "explicit feedback" can be intuitively fed back to the backend through the RLHF-Rete algorithm. From the perspective of implicit feedback, personalized products are mainly recommended based on user ratings and preferences. Fig. 3 illustrates the preference structure of implicit feedback.

In Fig. 3, user and order ratings are clustered, and preference results are analyzed based on product review results. A personalized recommendation list is generated using the RLHF-Rete algorithm, and the rules are dynamically configured and the recommendation list is updated technically, taking into account information such as time and environment. To ensure the alignment between marketing recommendations and user interest information, the Term Frequency-Inverse Document Frequency (TF-IDF) algorithm is proposed to calculate user interest by constructing IF-IDF vectors of sentence pairs to calculate the cosine distance between vectors. Eq. (4) is the mathematical expression of the IF-IDF vector.

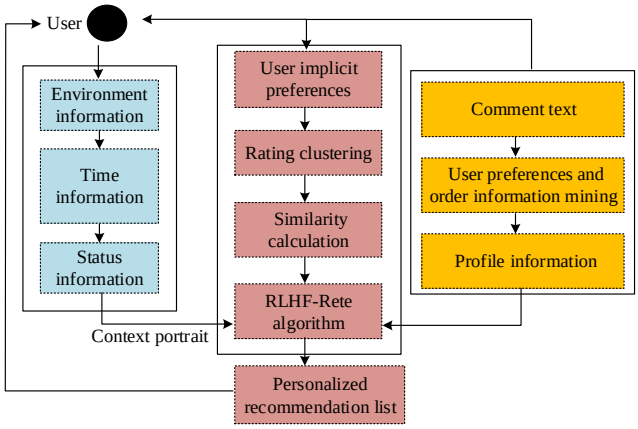


Fig. 3. Schematic diagram of preference structure for implicit feedback.

$$TF-IDF(w_i)=\frac{[Dt(w_i)]\times[\log|N|]}{[Num(SA\cup SB)]\times[\log(1+w_i:w_i\subset N)]} \quad (4)$$

In Eq. (4), $Dt(w_i)$ represents the frequency of occurrence of a word in a sentence, $Num(SA\cup SB)$ is the number of words shared between sentence A and sentence B, $|N|$ is the total number of sentences in the dataset, and w_i is the word [18]. With the help of this equation, the word can be changed to 'product' and the sentence can be changed to 'user', thus calculating the IF-IDF vector between the product and the user. Considering the subjective differences in user ratings, this study introduces vector mean to modify the cosine distance calculation, which calculates the correlation between users and products separately to obtain the mathematical expression of user interest under products, as shown in Eq. (5):

$$Pir_{x,g}=\sum_{y=1}^{q}\tilde{r}_{y,g}*sim(x,y)/\sum_{y=1}^{q}sim(x,y) \quad (5)$$

In Eq. (5), g represents the user, x, y are the products, q is the quantity of similar products, sim is the similarity between products, and $\tilde{r}$ is the user's rating of the products. The rating of different products can be achieved by adjusting the similarity weight, and Eq. (5) can be used to predict the level of interest of users in each product. Afterwards, based on Eq. (5), considering user similarity, the comprehensive formula for user interest is obtained, as shown in Eq. (6):

$$\begin{aligned}Pir_{x,g}'=\bar{r}_g+[\sum_{p=1}^{j1}(\tilde{r}_{x,p}-\bar{r}_p)*sim(g,p)/\sum_{p=1}^{j1}sim(g,p)]\\+[\sum_{y=1}^{j2}(\tilde{r}_{y,g}-\bar{r}_g)*sim(x,y)/\sum_{y=1}^{j2}sim(x,y)]\end{aligned} \quad (6)$$

In Eq. (6), $\bar{r}_g$ and $\bar{r}_p$ are the average ratings of all products under user g and p, while $j1$ and $j2$ are the sets of similar users and similar products. Considering that the training dataset designed for research is mostly static analysis content, to ensure the accuracy of predicting user preferences, the study proposes using a time sensitive function to solve the problem of user interest drift. The time function is defined as Eq. (7):

$$f(t)=e^{-\tau\cdot t} \quad (7)$$

In Eq. (7), t is time, e is the base of the natural logarithm, and τ is the time weight. The interest equation that introduces time weight can be expressed as Eq. (8):

$$Pir_{x,g}''=\frac{\sum_{t=1}^{j'}(\tilde{r}_{x,t}-\bar{r}_t)\times\partial(sim(x,y)\times\beta e^{-|t_x-t_y|})\times\tilde{r}_{x,s}\times\beta e^{\frac{t_r-t_{\min}}{t_{\max}-t_{\min}}}}{\sum_{t=1}^{j'}sim(x,y)} \quad (8)$$

In Eq. (8), $\partial()$ represents the logistic activation function, t_x, t_y represent the product rating time in the record set, $t_{\max}$, $t_{\min}$ are the maximum and minimum rating time, and t_r is the user's rating time. By using event sets, users' purchase records and ratings can be recorded, and similarity and similarity matrices can be calculated based on product feature data. Afterwards, the similarity matrix can be used to select the closest content that best matches the product and user. The remaining backup content can be calculated for interest and sorted in descending order, and merged with the previous real-time recommendation result to remove duplicates until the final result that meets the user's interest is obtained.

B. Control System Based on the Integration of Big Data Analysis and Machine Learning

Research uses the Apache Flink distributed streaming data processing framework to process and analyze various types of data. By constructing intelligent marketing systems at various dimensions and recommendation algorithms, the interactive experience between the system and customers can be achieved, including providing differentiated products and services, maintaining customer sources, and improving customer loyalty. The core of the Flink framework is a distributed streaming data engine written in Java and Scala, which can perform stateful computation on unbounded and bounded data streams, including simultaneous processing of batch and stream processing tasks, providing event time and processing time semantics, handling out of order event streams, as well as high throughput, low latency, state consistency assurance, and support for multiple running modes. It can adapt to different application scenarios and requirements. The Flink framework can adapt to different levels of Application Programming Interface (API) layers, and its structure includes a deployment layer, a Runtime core layer, an API layer, and an application layer [19]. Flink supports three temporal semantic expressions for real-time data processing: Event Time, Processing Time, and Ingestion Time. The study added the Watermark mechanism in Flink to solve the problem of temporal semantic disorder in Event Time. The combination of Watermark and Window can trigger a delay mechanism to ensure that the system can automatically verify data and meet time conditions. Fig. 4 shows the system operation process of Flink.

In Fig. 4, the client moves the data on the application to the cluster manager for resource balancing and scheduling, and hands over the creation of jobs to the application manager for resource loading and job management. Flink generates multiple task managers, which dynamically allocate tasks based on task slots and execute them through job graphs. Using Flink as the

streaming processing engine for marketing systems, program code conversion of syntax structures can be achieved, and combined with improved matching algorithms, the overall system performance can be improved, ensuring data transmission and output efficiency while enhancing intelligent marketing effectiveness [20]. When designing an intelligent marketing system, research is conducted on e-commerce platforms, analyzing business requirements, functional requirements, and non-functional requirements. In the part of business requirements, it is necessary to find the users who meet the conditions based on the image tag library and time span, and put the query messages into the cache system; in the part of functional requirements, it starts from the aspects of cache management, rule management, rule monitoring, and rule analysis; in the part of non-functional requirements, it starts from the consideration of the system's scalability, operation reliability, and data confidentiality. Fig. 5 is a schematic diagram of the core architecture design of the marketing system.

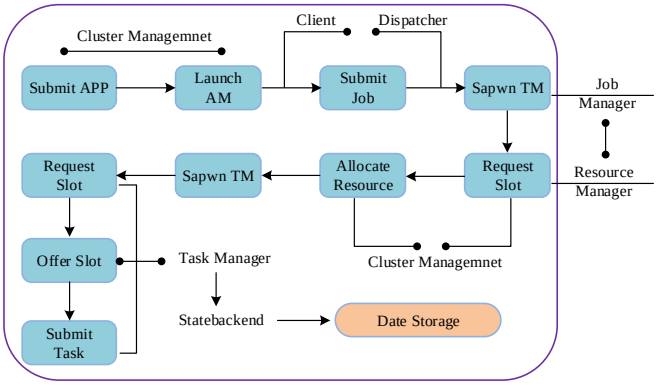


Fig. 4. Flink system operation process.

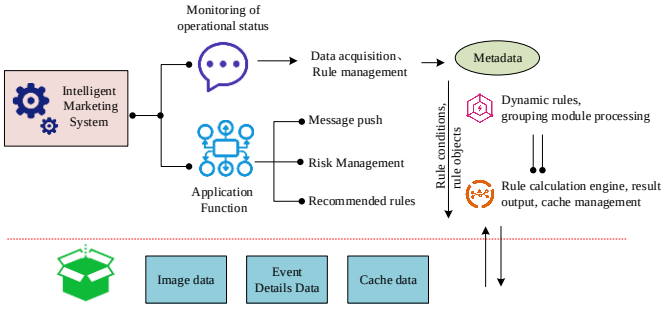


Fig. 5. Schematic diagram of the core architecture design of the marketing system.

In Fig. 5, the core architecture can be divided from top to bottom into rule management module, rule analysis module, application layer, data storage layer, and data caching layer. The data of the marketing system is mainly stored on the big data development platform, including collected user behavior data, business data, and other content. The rule analysis module utilizes collected user data to perform tasks such as analysis, calculation, and matching. In the database design part of the marketing system, it is necessary to design a data tag system related to users, including basic user attributes, activity status, interaction behavior, consumption ability, etc., and store user profile tag data in Hbase, using a two-level indexing method to achieve fast data retrieval [21]. The study was carried out with the help of IntelliJ IDEA 2021.2.3 development and VMware Workstation Pro for the configuration of the system cluster, and set up the auxiliary tools for each part, time tools, manager, event memory for the detailed design of each module.

IV. Results

A marketing digital control system was proposed based on big data analysis and machine learning algorithms, and refined from the perspectives of system function design and requirement analysis. The system control technology parameters used for the experiment were set up as follows: Intel (R) Core (TM) i7-1255H system (16GB memory, 512GB solid-state drive, Intel Iris Xe Graphics integrated graphics card, CentOS 6.3 operating system), data simulator Zookeeper 3.4.6.tar.gz, rule engine Flink, Drools (7.23.0 version), Kafka (2.11-20.0 version), and service cluster tool Hadoop. Considering that there is no standard publicly available dataset for analysis, the study aims to obtain data from two aspects: data policy and real data crawling to construct the dataset (consisting of over 30000 pieces of data). Among them, in the simulation data, a new application logic structure is added to the requirement, and a rule tree is established to achieve rule query and matching, using probability distribution to generate structured data. At the same time, using Zookeeper 3.4.6 to coordinate the state synchronization of distributed data generation nodes, ensuring data consistency; Simulate user behavior event streams (such as clicks, purchases, etc.) using Kafka, while Hadoop clusters store large-scale generated user profiles and behavior logs, supporting subsequent batch analysis. The triggering conditions include user profile attributes and behavior attribute conditions, where the user profile includes gender, age, income, occupation, region, etc., and the behavior attribute conditions include consumption behavior, product browsing status, comment status, etc. Analyze the facts, conditions, thresholds, and time elements based on rule conditions. In the real data acquisition section, the study utilized web crawling tools to obtain legitimate marketing data on platforms such as Taobao and JD.com, including consumption patterns, after-sales ratings, etc. A total of 13256 comments were collected. Firstly, the matching results of the improved algorithm proposed by the research under the rule set were analyzed, and the matching accuracy results are shown in Fig. 6.

Fig. 6(a) shows the accuracy and time consumption of traditional algorithms. The traditional algorithm reached its maximum matching accuracy of 62.15% when the dataset was 6000. However, as the number of rules increased, its matching accuracy showed a downward trend, with a maximum time consumption of 5.4 seconds. Fig. 6(b) shows the accuracy and time consumption of the improved algorithm, where the matching accuracy curve of the improved algorithm shows an upward trend, with a maximum matching accuracy of 90.12%, and is relatively less affected by the number of rule sets, with a time consumption of no more than 3 seconds. Fig. 6(c) shows the product matching results under the influence of time weight. Changes in the time weight coefficient will correspondingly affect the recommendation effect. When the coefficient value was 0.3, the similarity calculation results

performed better. The improved algorithm exhibited higher recommendation accuracy compared to traditional algorithms, with an improvement of over 5% in accuracy. Afterwards, the improved algorithm was compared with Recursive Flow Classification (RFC), Spatial Structure Matching (SSM), and TextRank-Hierarchical Clustering (TextRank-HC) algorithms to analyze the rule matching results under different rule numbers and matching times. The comparison results are shown in Fig. 7.

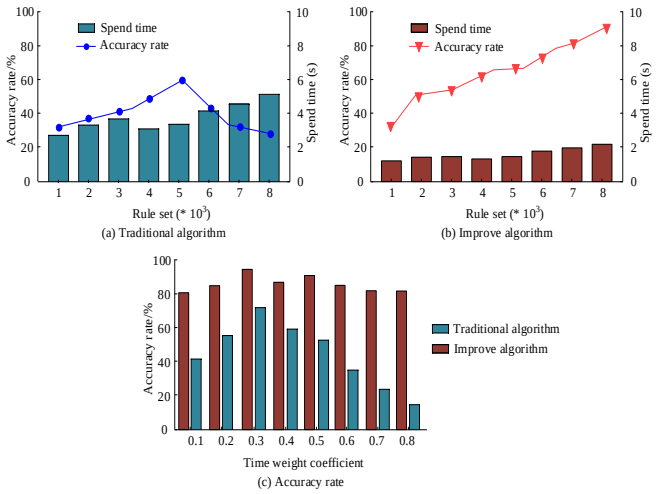


Fig. 6. Improved algorithm for matching results under rule sets.

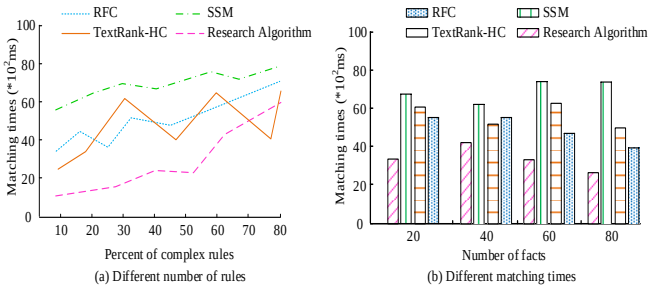


Fig. 7. Rule matching results under different numbers of rules and matching times.

In Fig. 7, under different proportions of complex facts, the improved algorithm had the shortest matching time, with a maximum value not exceeding 6000ms, and the overall spending time showed a relatively slow growth trend as the number of facts increased. Secondly, the RFC algorithm and TextRank-HC algorithm, which require less matching time, had a maximum processing time of no more than 7500ms. The SSM algorithm performed the worst and had a significant growth rate. Under different complex rules, the ranking of algorithms with the highest matching time was: SSM algorithm>TextRank-HC algorithm>RFC algorithm>research algorithm, and the research algorithm had the lowest growth rate of matching duration, with good real-time application and obvious response effect. Based on the application of the algorithm proposed by the research in marketing systems, this study analyzed the prediction of user consumption behavior and the effectiveness of product recommendations, and compared it with the K-Prototype Clustering Algorithm (K-prototype), the Clustering by Fast Search and Find of Density Peaks (CFSFDP), and the Comment perception Graph Neural Networks (RGRS). Fig. 8 to Fig. 9 shows the confusion matrices of different algorithms for predicting user consumption behavior.

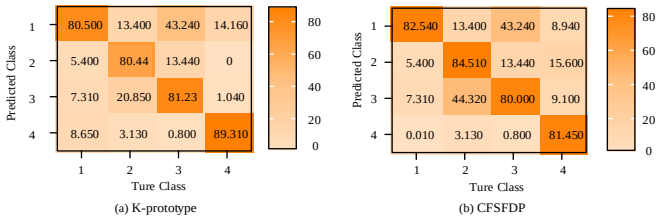


Fig. 8. The confusion matrix of K-prototype algorithm and CFSFDP algorithm for predicting user consumption behavior.

Fig. 8 shows the consumption behavior prediction results of the K-prototype algorithm and CFSFDP algorithm for four types of consumer goods (essential goods category-1, electronic products category-2, clothing category-3, and food category-4). The results showed that the prediction accuracy of the K-prototype algorithm exceeded 80%, and the prediction accuracy in food consumption behavior reached 89.31%. The prediction results of CFSFDP were slightly worse than those of the K-prototype algorithm, but it exceeded 80%. Afterwards, the prediction results of the RLHF-Rete algorithm and the proposed algorithm were analyzed, as shown in Fig. 9.

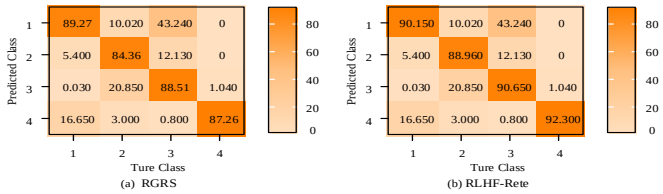


Fig. 9. The confusion matrix of the RLHF-Rete algorithm and the research algorithm for predicting user consumption behavior.

In Fig. 9, the RLHF-Rete algorithm had a prediction accuracy of over 84% for user consumption behavior, while the proposed algorithm had a prediction accuracy of over 88%. The recommendation performance was significant, and the deviation of its recommendation results was relatively small for the four types of products. The above algorithm was subjected to statistical significance testing, and the results are shown in Table I.

In Table I, the RFC algorithm suffers from accuracy loss due to recursive decomposition in terms of prediction accuracy metrics ($p<0.001$, effect size 1.87); The spatial structure calculation of SSM brings significant time overhead (34.8% more matching time); The text feature extraction and clustering of TextRank HC suffer from information loss (accuracy difference of 21.9%); The RLHF Rete algorithm can effectively compensate for information loss and has better prediction accuracy than other algorithms. Afterwards, the application effect of the above algorithm in marketing intelligent control was analyzed, and the results are shown in Fig. 10.

TABLE I. STATISTICAL TEST RESULTS OF DIFFERENT ALGORITHM PERFORMANCE

Evaluation indicators	RLHF-Rete	Comparison algorithm	*p* value	Effect size(Cohen's d)
Accuracy of consumer behavior prediction/ %	88.00±1.2	RFC:76.35±3.2	<0.001	1.87
		SSM:68.42±4.1	<0.001	2.45
		TextRank-HC:72.18±2.9	<0.001	2.01
MAE	0.0712±0.005	RFC:0.1521±0.011	<0.001	2.78
		SSM:0.1843±0.015	<0.001	3.12
		TextRank-HC: 0.1368±0.010	<0.001	2.35
F1 value	0.6923±0.015	RFC:0.5824±0.022	<0.001	1.98
		SSM:0.5217±0.025	<0.001	2.67
		TextRank-HC:0.6035±0.020	<0.001	1.45
Response delay (ms)	5820±210	RFC:7260±315	<0.001	1.89
		SSM:8920±405	<0.001	2.98
		TextRank-HC:8140±380	<0.001	2.45

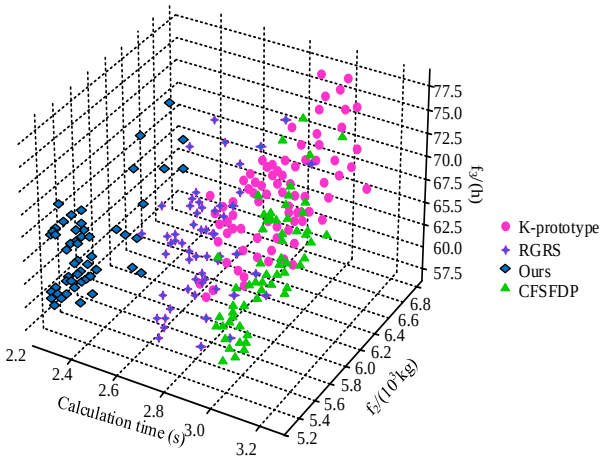


Fig. 10. System testing time results.

In Fig. 10, the research algorithm had a lower computational time compared to other comparative algorithms, with an overall time of no more than 2.2 seconds, and it also performed well in the other two indicators, with a more concentrated overall distribution. However, the performance of the K-prototype algorithm was poor, and the system application effect was not significant. Further analysis was conducted on the application effect of the above algorithms in marketing digitization, and the results are shown in Table II.

TABLE II. PERFORMANCE OF MARKETING DIGITALIZATION APPLICATIONS WITH DIFFERENT ALGORITHMS

Evaluation indicators	RLHF-Rete	K-prototype	CFSFDP	RGRS
F1 value	0.8923±0.015	0.7412±0.021	0.7849±0.018	0.8348±0.017
Response delay (ms)	5820±210	8240±355	7910±320	6850±280
Cold start adaptability (%)	83.5±2.1	72.3±4.2	68.7±3.8	78.6±2.9
Rule interpretability (rating)	4.8/5.0	3.2/5.0	2.9/5.0	2.1/5.0
Coverage rate of long tail products (%)	78.2±3.1	62.5±5.4	58.3±4.8	71.6±4.2
Compute Resource Consumption (GPU-h)	12.5±1.2	8.7±0.9	7.9±0.8	18.3±1.5

In Table II, the recommended F1 values for RLHF Rete algorithm, K-prototype algorithm, CFSFDP algorithm, and RGRS algorithm are 0.8923, 0.7412, 0.7849, and 0.8348, respectively. In the cold start scenario, the RLHF Rete algorithm can achieve a higher click through rate for product recommendations than other algorithms through rule templates and small sample learning. Additionally, its Gini coefficient for long tail product coverage is 0.21, which is much lower than that of the CFSFDP algorithm (0.38), reflecting a more equitable distribution of traffic. The GPU consumption of RLHF Rete algorithm is higher than that of clustering algorithm, but its unit request processing cost is 31.7% lower than that of RGRS algorithm. Afterwards, the marketing digital control system proposed by the research was tested and analyzed, and the results are shown in Table III.

TABLE III. MODULE TESTING AND PERFORMANCE TESTING RESULTS OF THE CONTROL SYSTEM

Project			Before application	After application
Module testing	Cache management	Data connection success rate	87%	94%
		Data query efficiency error	7%	2%
		Query time consumption	2.5s	0.8s
	Rule management	Sensitivity triggered by conditions	Medium	High
		Dynamic rule matching	80%	95%
		Dynamic rule modification	10min	3min
		Display sensitivity	Commonly	Excellent
	Rule monitoring	Data access	Manual	Automatic
		Time triggered event	Instable	Stable
		Execution status	75%	92%

	Rule analysis	Data collection and generation efficiency	82%	91%
		Server response	3.0s	0.7s
		Code running status	Occasionally make mistakes	Stable
Performance testing	Static environment	Data processing time	50s	10s
		Load balancing rate	79%	88%
		CPU usage rate	80%	45%
		Memory utilization rate	70%	50%
		Request (1000 times) query event takes time	30s	5s
		Overall stability	Generally	High
	Dynamic environment	Data processing time	95s	15s
		load balancing	72%	86%
		CPU usage rate	90%	50%
		Memory utilization rate	85%	60%
		Request (1000 times) query event takes time	40s	8s
		Overall stability	Generally	High

In Table III, the control system designed had good data analysis capabilities, with a data query time of only 0.8 seconds and a data collection and generation efficiency of over 90%. It could effectively improve resource allocation efficiency and adapt to system applications in both static and dynamic environments, with relatively stable performance. Afterwards, the application effect of the marketing system was analyzed, and the results are shown in Table IV.

TABLE IV. APPLICATION EFFECT OF MARKETING DIGITAL CONTROL SYSTEM

Project		Before application	After application
Application effect	Marketing conversion rate	2%	6%
	Clue follow-up efficiency	50%	90%
	Customer retention rate (6 months)	30%	50%
	Order conversion rate	10%	15%
	User satisfaction rating	7.5/10	9.0/10

In Table IV, the control system designed could predict consumer behavior, provide personalized recommendation services, timely follow up on 90% of leads, achieve a marketing conversion rate of over 5%, and improve order conversion rates and user satisfaction ratings to varying degrees.

V. DISCUSSION

The emergence of the Internet platform and the development of science and technology information technology make it easy for users to obtain massive data and realize online shopping [22]. Traditional enterprise marketing uses data warehouses to achieve information protection for user groups, and uses a single recommendation algorithm to predict user preferences. However, it ignores the real-time and interactive nature of the marketing system, which makes its application effect limited and difficult to meet the needs of enterprises [23]. The marketing system covers various aspects such as education, social interaction, consulting, life, and finance. Its core can be explained through three modules: input, recommendation, and output, which essentially involve the exchange of information services or products [24]. Accelerating the digitalization process of enterprises and the digital transformation of industries is an important aspect of promoting the development of the digital economy [25]. The existing marketing recommendation system, when using product user similarity to promote sales, fails to fully consider the preferences and needs of each user, with few feedback methods and mostly staying at the level of one-sided product recommendations [26]. To enhance the dynamic interaction between the system and users, research proposes rule-based algorithms and machine RL recommendation algorithms to achieve personalized recommendations.

The results showed that the improved algorithm proposed in the study exhibited a continuously improving matching accuracy as the number of rule sets increased, with a maximum matching accuracy of 90.12%, while the maximum matching accuracy of traditional algorithms was only 56.24%. When the time weight coefficient value was 0.3, the similarity calculation results performed well, and the improved algorithm has improved recall and accuracy by more than 5% compared to traditional algorithms. Under different proportions of complex facts, the rule matching time of the improved algorithm was the shortest, with a maximum value not exceeding 6000ms, and the growth trend was relatively slow. Next were the RFC algorithm and the TextRank HC algorithm, with a maximum time consumption of no more than 7500ms. SSM algorithm performed the worst and had a significant growth rate. The ranking of algorithms with the highest matching time under different complex rules as: SSM algorithm>TextRank-HC algorithm>RFC algorithm>research algorithm. The RLHF Rete algorithm dynamically optimizes the rule network structure through reinforcement learning, reducing redundant rule matching paths and keeping the matching time within 6000ms. The SSM algorithm needs to handle spatial structural similarity calculations, while TextRank HC has a high time complexity due to simultaneous text analysis and hierarchical clustering. Although the RFC algorithm adopts recursive optimization, it does not solve the problem of rule conflicts,

making its matching performance second only to research methods. In terms of predicting user consumption behavior, the accuracy values of the three compared models all exceeded 80%, but they were slightly inferior to the RLHF-Rete algorithm proposed by the research. The RLHF-Rete algorithm had a prediction accuracy of over 88% for user consumption behavior, and the overall deviation was relatively small. 88% of the accuracy of consumer behavior prediction comes from the effective integration of structured behavior data and unstructured comments in the research method. Its RLHF mechanism can automatically adjust the threshold, which is more flexible than the CFSFDP fixed density threshold. Compared to the fuzzy control in reference [6], the proposed RLHF Rete improves real-time performance from a minute level to a second level while maintaining comparable interpretability of rules. And compared with the association rules and Apriori algorithm in reference [11], the research method can maintain linear growth through rule pruning. The product recommendation results showed that the MAE value (0.0712) exhibited by the RLHF-Rete algorithm was smaller than the other three comparison models. Among them, the K-prototype algorithm had the largest MAE value, reaching 0.1742, followed by the CFSFDP algorithm. The F1 value of the RLHF-Rete algorithm (0.6923) was greater than that of the K-prototype algorithm (0.6212), CFSFDP algorithm (0.5849), and RGRS algorithm (0.6348). The application effect of marketing intelligent control showed that the research algorithm had less computation time than other comparative algorithms, with an overall time of no more than 2.2 seconds. The data query time of this control system was only 0.8 seconds, and its data collection and generation efficiency exceeded 90%. It could effectively improve resource allocation efficiency and adapt to system applications in static and dynamic environments. Moreover, it could predict consumer behavior through marketing, with 90% of leads being promptly followed up, a marketing conversion rate exceeding 5%, and an order conversion rate exceeding 10%.

VI. Conclusion

The marketing digital control system based on the integration of big data analysis and machine learning proposed by the research has good application effects, which can effectively predict user consumption behavior and achieve high accuracy in product matching. Although the marketing digital control system designed in this study performs well in matching accuracy, recommendation effectiveness, and real-time performance, the system still heavily relies on large-scale user behavior data. However, in reality, there are data sparsity or noise interference that may affect the stability of reinforcement learning rule generation. The Apache Flink framework design may increase rule dynamic latency in high concurrency scenarios, resulting in lagging recommendation results and insufficient balance between personalization and generalization. Therefore, in future improvement work, research attempts to increase multimodal learning and deploy incremental learning frameworks to better model user product social relationships and adapt to data distribution drift. At the same time, by combining knowledge graphs and considering factors such as group behavior patterns and social network dissemination effects on marketing recommendations, more rules are extended to generate content, and a hierarchical rule engine is designed to deploy lightweight recommendation models to enhance the application effectiveness of the model system.

References

[1] Kopalle P K, Gangwar M, Kaplan A, Ramachandran D, Reinartz W, Rindfleosch A. Examining artificial intelligence (AI) technologies in marketing via a global lens: Current trends and future research opportunities. International Journal of Research in Marketing, 2022, 39(2): 522-540.

[2] Plangger K, Grewal D, de Ruyter K, Tucker C. The future of digital technologies in marketing: A conceptual framework and an overview. Journal of the Academy of Marketing Science, 2022, 50(6): 1125-1134.

[3] Ye Z, Zhang D J, Zhang H, Zhang R, Chen X, Xu Z. Cold start to improve market thickness on online advertising platforms: Data-driven algorithms and field experiments. Management Science, 2023, 69(7): 3838-3860.

[4] Abrardi L, Cambini C, Rondi L. Artificial intelligence, firms and consumer behavior: A survey. Journal of Economic Surveys, 2022, 36(4): 969-991.

[5] Nguyen D K, Sermpinis G, Stasinakis C. Big data, artificial intelligence and machine learning: A transformative symbiosis in favour of financial technology. European Financial Management, 2023, 29(2): 517-548.

[6] Ikegwu A C, Nweke H F, Anikwe C V, Alo U R, Okonkwo O R. Big data analytics for data-driven industry: a review of data sources, tools, challenges, solutions, and research directions. Cluster Computing, 2022, 25(5): 3343-3387.

[7] Franke M, Pulignano V. Connecting at the edge: Cycles of commodification and labour control within food delivery platform work in Belgium. New Technology, Work and Employment, 2023, 38(2): 371-390.

[8] Hicham N, Nassera H, Karim S. Strategic framework for leveraging artificial intelligence in future marketing decision-making. Journal of Intelligent Management Decision, 2023, 2(3): 139-150.

[9] Liu Q, Guo L. Transformation and development strategy of digital publishing marketing based on big data and fuzzy control algorithm. Journal of Computational Methods in Sciences and Engineering, 2024, 24(1): 489-499.

[10] Rivas P, Zhao L. Marketing with chatgpt: Navigating the ethical terrain of gpt-based chatbot technology. AI, 2023, 4(2): 375-384.

[11] Liao S H, Yang C A. Big data analytics of social network marketing and personalized recommendations. Social Network Analysis and Mining, 2021, 11(1): 21.

[12] Cherkaoui N, Handri K E, Medard D Y T, El Hassani Y, Errafyg A. Consumer behaviour: analysing marketing campaigns through recommender systems and statistical techniques. Marketing i menedžment innovacij, 2024, 15(3): 1-12.

[13] Iyelolu T V, Agu E E, Idemudia C, Ijomah T I. Leveraging artificial intelligence for personalized marketing campaigns to improve conversion rates. International Journal of Engineering Research and Development, 2024, 20(8): 253-270.

[14] Rithani M, Kumar R P, Doss S. A review on big data based on deep neural network approaches. Artificial Intelligence Review, 2023, 56(12): 14765-14801.

[15] Nalbant K G, Aydın S. Development and transformation in digital marketing and branding with artificial intelligence and digital technologies dynamics in the Metaverse universe. Journal of Metaverse, 2023, 3(1): 9-18.

[16] Lopez S, Arjunan G. Optimizing marketing ROI with predictive analytics: Harnessing big data and AI for data-driven decision making. Journal of Artificial Intelligence Research, 2023, 3(2): 9-36.

[17] Paramesha M, Rane N L, Rane J. Big data analytics, artificial intelligence, machine learning, internet of things, and blockchain for enhanced business intelligence. Partners Universal Multidisciplinary Research Journal, 2024, 1(2): 110-133.

[18] Andronie M, Iatagan M, Uță C, Hurloiu I, Dijmărescu A, Dijmărescu I. Big data management algorithms in artificial Internet of Things-based fintech. Oeconomia Copernicana, 2023, 14(3): 769-793.

[19] Lăzăroiu G, Bogdan M, Geamănu M, Hurloiu L, Luminita L, Ștefănescu R. Artificial intelligence algorithms and cloud computing technologies in blockchain-based fintech management. Oeconomia Copernicana, 2023, 14(3): 707-730.

[20] Mokayed H, Quan T Z, Alkhaled L, Sivakumar, V. Real-time human detection and counting system using deep learning computer vision techniques[C]//Artificial Intelligence and Applications. 2023, 1(4): 221-229.

[21] Barzizza E, Biasetton N, Ceccato R, Salmaso L. Big Data Analytics and Machine Learning in Supply Chain 4.0: A Literature Review. Stats, 2023, 6(2): 596-616.

[22] Shah H M, Gardas B B, Narwane V S, Mehta H S. The contemporary state of big data analytics and artificial intelligence towards intelligent supply chain risk management: a comprehensive review. Kybernetes, 2023, 52(5): 1643-1697.

[23] Gallego V, Lingan J, Freixes A, Juan A A, Osorio C. Applying Machine Learning in Marketing: An Analysis Using the NMF and k-Means Algorithms. Information, 2024, 15(7): 368.

[24] Li L, Yuan L, Tian J. Influence of online E-commerce interaction on consumer satisfaction based on big data algorithm. Heliyon, 2023, 9(8): e18322.

[25] Adeniran I A, Efunniyi C P, Osundare O S, Abhulimen A O. Transforming marketing strategies with data analytics: A study on customer behavior and personalization. International Journal of Management & Entrepreneurship Research, 2024, 6(8): 41-51.

[26] Amudharasan A. The impact of recommendation algorithms: Analyzing the influence of data on marketing strategies in the media sector. Open Journal of Business and Management, 2023, 11(6): 3373-3384.

Artificial Intelligence in Optometry: Potential Benefits and Key Challenges: A Narrative Review

Noura A. Aldossary
Department of Optometry, King Saud University, Riyadh, Saudi Arabia

***Abstract*—The integration of Artificial Intelligence (AI) into healthcare is transforming many medical fields, including optometry. This study provides a narrative review of the current applications and future potential of AI in optometric practice, emphasizing its role in automated screening and diagnosis, personalized treatment planning, and enhanced accessibility through tele-optometry. Alongside these opportunities, this study examines the technical, socioeconomic, ethical, legal, and professional challenges that limit the effective integration of AI in optometry practice. Focus is placed on concerns surrounding data privacy, patient autonomy, regulatory disparities, and practitioner resistance to adoption. Furthermore, this review highlights key research gaps, including the need for diverse training datasets, large-scale validation trials, and collaborative training between clinicians and AI developers. By resolving these challenges, AI has the potential to improve diagnostic accuracy, expand access to care, and enhance the quality of eye care services. By integrating the available evidence, this narrative review provides clinicians, policymakers, and researchers with a comprehensive overview of the benefits, challenges, and future directions of AI in optometry.**

Keywords*—*Artificial intelligence; automated screening; tele-optometry; eye care services; patient autonomy

I. Introduction

Artificial Intelligence (AI) is the ability of computer systems to perform tasks that normally require human intelligence, such as decision-making, pattern recognition, and problem-solving [1]. These processes include learning (the acquisition of information and rules for using this information), reasoning (the use of rules to arrive at precise or imprecise conclusions), and self-correction [2].

Machine Learning (ML), a subdivision of AI, primarily utilizes computer programming to execute tasks or forecast outcomes [3]. ML has shown significant promise in clinical decision support and machine translation [4]. Conventional ML algorithms rely on expert-selected variables as inputs and typically do not involve extensive neural networks. These algorithms include linear and logistic regression, support vector machines, decision trees, and random forests [5].

Deep Learning (DL), a subcategory of ML, can automatically derive rules from known data to evaluate unknown data without specialized programming, thus enabling it to handle more intricate data [6]. DL algorithms generally incorporate large-scale neural networks, including artificial, convolutional, and recurrent neural networks [5]. Recent advancements in AI, particularly ML and DL, have influenced the healthcare industry by increasing diagnostic efficiency, optimizing treatment, and lowering costs [7]. Fig. 1 shows the hierarchical relationship between AI, ML, and DL.

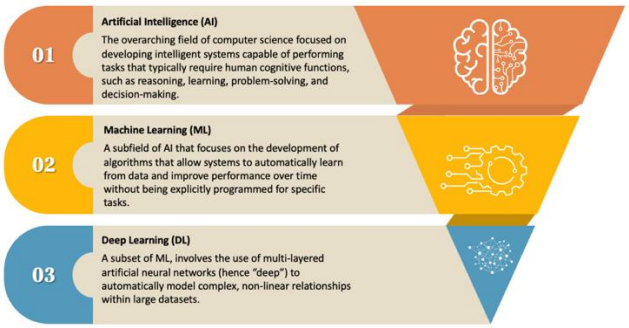


Fig. 1. Hierarchical relationship between Artificial Intelligence, Machine Learning and Deep Learning [3].

AI has demonstrated significant potential in clinical practice [8], particularly for improving diagnostic accuracy. The use of AI algorithms in the diagnosis of diseases such as cancer, heart disease, and neurological disorders has achieved an accuracy of 95% [9]. According to Ali et al. (2024), AI-powered algorithms demonstrated superior performance in detecting pneumonia from chest X-rays, achieving an accuracy rate of 92.7%, surpassing the diagnostic capabilities of human radiologists [10].

II. Background

The term "optometry" is derived from the Greek words "optos" meaning 'see' and "metron" meaning 'measure' and is concerned with the diagnosis and treatment of eye diseases and impaired vision. The field has undergone a significant transformation over the past century; in the early 1900s, visual optics was largely considered a philosophical study of light and vision.

By the mid-20th century, optometry had evolved into a clinical field, focusing on refractive errors and other ocular disorders, with corrective lenses serving as the primary method of management [11], [12]. Optometry practices manage a wide range of refractive and visual conditions, including myopia (near-sightedness), hyperopia (farsightedness), presbyopia (age-related farsightedness), astigmatism (vision distortion caused by an irregular corneal shape), anisometropia (unequal focal power between eyes), strabismus (squinting), and amblyopia (lazy eye) [11].

The integration of AI into optometry represents a game-changing development that fundamentally alters the approaches used by eye care specialists to detect and manage various eye

disorders. This technological advancement is reforming optometric practice as the profession evolves, emerging as a transformative force in the diagnosis, treatment and management of diverse ocular conditions. By leveraging advanced algorithms and DL models, AI has introduced innovative approaches that enhance diagnostic accuracy and patient management and cover the system for personalized treatment, ultimately reshaping the future of eye care. According to a systematic review and meta-analysis published in 2024, the global incidence of myopia is anticipated to exceed 740 million cases by 2050 [13], with a significant increase particularly observed in the urban areas of East and Southeast Asia. Myopia currently affects approximately 30% of the population in North America and Europe, with rates approaching 80-90% in some parts of East Asia [14]. With the global prevalence of vision impairment increasing, an estimated 2.2 billion people are affected by some form of visual impairment worldwide [13], [14]. This trend has significantly increased the demand for innovative diagnostic and treatment methods in optometry [15]. One notable application of AI in this field is automated retinal image analysis, where algorithms can detect certain retinal diseases, such as diabetic retinopathy, with high sensitivity and specificity [16].

This study examines the opportunities and risks associated with the application of AI in optometry, as well as its practical implications. AI-driven technologies, such as diagnostic tools, treatment planning systems, and tele-optometry, are increasingly being integrated into optometric practice, offering the potential to improve the accuracy of diagnosis, quality of patient outcomes, and accessibility to eye healthcare.

III. Application of AI in Optometry

The role of AI in detecting, diagnosing, and treating several vision-related complications cannot be overstated, with the help of advanced ML and DL techniques. This section outlines the key applications of AI in optometry.

A. Diagnostic Tools and Image Analysis

AI has transformed diagnostic tools in optometry, particularly in the field of imaging. The automated diagnosis of retinal diseases using image recognition through AI systems has become a topic of interest. A systematic review and meta-analysis by Islam et al. (2020) indicated that DL algorithms trained on retinal fundus images can diagnose diabetic retinopathy with a sensitivity of 83% and a specificity of 92% [17].

Furthermore, AI has been utilized to identify age-related macular degeneration (AMD) and classify its stages by analyzing Optical Coherence Tomography scans with high accuracy [18]. The applications of DL to Optical Coherence Tomography in AMD were explored in a systematic review and meta-analysis conducted by Paul et al. [19]. They reported various applications, including disease diagnosis and classification, segmentation of retinal layers and biomarkers, prediction of disease progression and visual function, and determining the requirement for referral to a retinal specialist, with a performance comparable to that of human specialists.

The pattern recognition capabilities of DL have also been used to detect differences in corneal topographic patterns between normal individuals and those with keratoconus. For example, Kuo et al. proposed three Convolutional Neural Network (CNN) models, namely VGG16, InceptionV3, and ResNet152 [20]. These models were trained using topographic images without manual segmentation. All models achieved sensitivity, specificity, and area under the receiver operating characteristic curve (AUROC) values exceeding 90%, with the ResNet152 model reaching an AUROC of 0.995, indicating the high diagnostic accuracy of all models, with ResNet152 being superior. These results demonstrate the potential of AI-based approaches to enhance the detection and classification of keratoconus.

Moreover, ML and DL techniques have been applied to diverse image-based diagnostic and predictive solutions for myopia. AI tools have been developed for the detection, diagnosis, and prediction of myopia progression in children and adults [21], [22]. For instance, “DeepMyopia” is a DL system that was developed to detect and predict myopia onset in children at risk using retinal fundus images [23]. The system was trained on more than 1.6 million images and validated on different datasets from seven sites in China. The model showed a robust predictive performance, with AUCs ranging from 0.81–0.91 for 1- to 3-year onset prediction. DeepMyopia also successfully classified children into low- and high-risk groups, demonstrating its ability as a decision support tool.

Research has highlighted strabismus as another domain in which AI can provide valuable support. In 2021, a team led by Mao et al. established an AI-based platform that incorporated three DM systems [24]. This platform was designed to diagnose strabismus, assess angles, plan surgical interventions, and utilize corneal light-reflection images. The researchers trained and retrospectively validated the system using a historical development dataset. The results demonstrated that the AI-based screening method exhibited remarkable performance, with a sensitivity of 99.1%, specificity of 98.3%, area under the receiver operating characteristic curve (AUC) of 0.998, and matching accuracy of 99.0% in a retrospective evaluation. These metrics were comparable to or exceeded those of previously established automated techniques for strabismus identification. Building on such advancements, another study reported the development of a novel DL algorithm designed to objectively categorize eye versions from photographs of adult faces using a mobile application [25]. The model underwent initial training at nine gaze positions. Depending on the specific type of eye version, the application demonstrated an accuracy of 42 to 92% and precision of 28 to 84%. In a related effort, Zheng et al. developed an enhanced DL model trained on 7,026 images of children with primary horizontal strabismus [26]. The algorithm was evaluated on 277 images from a different dataset, achieving 95% accuracy, exceeding the performance of resident ophthalmologists. These innovations have led to faster and more accurate diagnoses, reduced clinicians’ workloads, and improved the overall quality of care. They are particularly valuable in emergency rooms and mass-screening initiatives.

B. Improved Accessibility and Screening through Tele-optometry

Tele-optometry refers to the delivery of optometric care through telehealth technologies, allowing optometrists to

remotely evaluate and manage eye health [27]. AI-powered tele-optometry can facilitate the delivery of eye care services to underserved regions by analyzing patient-provided images, such as retinal scans and ocular images, to support early diagnosis and timely management [28]. During the COVID-19 pandemic in India, Karthikeyan et al. reported that 50.94% of optometrists relied on tele-optometry to provide their services, indicating its potential to remain a valuable component of eye care delivery beyond emergency settings [29]. Advances in DL further enhance this ability by enabling automated image analysis within tele-optometry platforms, supporting the remote management of posterior eye diseases, and referring cases to specialists when required, minimizing the need for face-to-face visits [30]. This technology not only improves the efficiency of eye care services but also alleviates the burden on overcrowded eye care clinics, allowing clinicians to prioritize urgent cases and minimize unnecessary referrals.

Another key advantage of AI-powered image analysis technology is its ability to remotely monitor chronic conditions, such as diabetic retinopathy and glaucoma. In particular, AI-based glaucoma screening can enhance the detection of moderate and advanced cases in primary care environments and help identify individuals at a higher risk of developing the condition [31]. This enables the early identification of disease progression, allowing timely intervention that may prevent further vision loss. Integrating AI with tele-optometry improves screening efficacy and optimizes the allocation of professional resources.

C. Personalized Care

AI supports individualized care in optometry, particularly in developing customized therapies for specific visual conditions. With access to vast pools of patient datasets, AI can develop tailored regimens based on visual characteristics, lifestyle, and other factors [11] [16]. Current AI-driven tools are used to design personalized contact lenses and eyeglasses by analyzing parameters such as visual acuity, astigmatism, eye structure, and even digital device usage patterns. This approach ensures more precise lens selection and recommendations, leading to effective and patient-centered solutions [32] [33].

AI is also advancing individualized care through the integration of CNNs in medical diagnostics. Hwang et al. trained an AI-based system to diagnose AMD using 35,900 Optical Coherence Tomography images, achieving over 90% accuracy, comparable to retinal specialists and outperforming medical students [34]. Beyond diagnosis, AI systems can provide appropriate treatment recommendations comparable to those of retinal specialists, demonstrating their potential for both remote and individualized care applications.

Moreover, AI systems show promise in complementing vision therapy, which is a core component of optometry care. Vision therapy involves structured exercises and activities designed to improve specific visual functions, such as eye movement coordination and binocular vision [35]. Vision therapy based on AI systems can continuously analyze patient data and assess their progress throughout the therapeutic process [36]. AI-enhanced vision therapy has the potential to optimize outcomes and improve patient adherence through continuous monitoring, follow-up care, and adaptive training. Fig. 2 provides a summary of the AI applications in optometry.

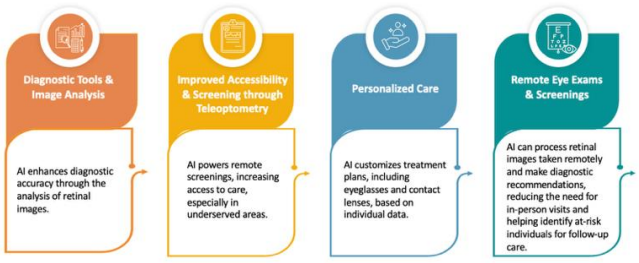


Fig. 2. Application of artificial intelligence in optometry.

III. Challenges of AI in Optometry

While AI presents significant opportunities for enhancing optometric care, its implementation in clinical practice is accompanied by several key challenges that must be overcome. These challenges extend beyond technical limitations, including socioeconomic issues, ethical considerations regarding patients' rights and data use, regulatory and legal complexities, and resistance from practitioners and patients. The following sections outline the main challenges that need to be addressed for the successful integration of AI in optometry, along with the current strategies that have been proposed to address them.

A. Technical Challenges

One of the major challenges in the application of AI in optometry is the methodological complexity of designing effective and precise models. DL-based AI systems, in particular, require extensive amounts of high-quality labeled data for effective training and performance [37]. Parmar et al. (2024) indicated that the performance of AI algorithms in identifying retinal diseases was greatly dependent on the quality of the training datasets. Models trained on biased or limited data may produce inaccurate or non-generalizable results when applied to diverse patient populations [38].

Another critical technical challenge is the "black box phenomenon", which refers to the lack of interpretability of AI models. Various DL models, particularly CNNs, are described as "black boxes" because their decision-making processes are challenging for clinicians to fully understand. This lack of transparency can undermine trust in AI systems, especially when they are used to make life-and-death clinical decisions [39]. To address this, explainable AI (XAI) strategies have been introduced to provide clearer insights into the decision-making process and enhance the overall transparency of AI models. For example, the OptiDex model was developed to enhance the detection and categorization of diabetic retinopathy. An XAI framework was incorporated using gradient-weighted class-activation maps. The model achieved a high diagnostic accuracy of 97.65% and provided visual heatmaps that explained how decisions were made [40]. This clarity improved clinician trust by indicating which retinal regions affected the output of the system. These examples emphasize how XAI can reduce the gap between accuracy and interpretability, improving both the confidence and adoption of AI in clinical optometry.

Furthermore, implementing and maintaining AI systems can be challenging because of their technical complexity, which

often requires specialized expertise and continuous support from the organization. Although it is crucial to smoothly incorporate these systems into existing clinical practices and electronic health record platforms, this integration process remains technically demanding.

B. Socioeconomic Challenges

The adoption of AI in optometry has emerged along with pressing socioeconomic challenges, particularly those associated with cost, accessibility, and healthcare equity. Among these, cost remains the most significant challenge, as AI integration involves expensive imaging machinery, advanced software, and considerable investments in staff training and resources. Muradov (2024) estimated the cost of the initial setup of the AI-based Retinal disease screening system to be between $30,000 and $50,000 [41]. Most of these costs are not covered by current reimbursement schemes; therefore, clinics in rural or low-income areas may not be able to afford such technologies. Continuous updates and maintenance contribute to the overall cost of implementing AI systems. The substantial upfront costs may place these technologies beyond the reach of smaller practices, potentially creating disparities between well-funded and under-resourced environments. To ensure the fair distribution of AI-enabled tools across all optometric practices, cost-reduction strategies, such as cloud-based alternatives or shared resources, should be explored.

Moreover, AI-based tele-optometry relies on a stable Internet connection and high-quality imaging, both of which are often limited in rural areas, limiting the application of AI in such regions [42]. Another key issue is the underrepresentation of specific population groups in the training datasets, which can result in biased AI algorithms and, therefore, provide less accurate outcomes for these groups. Such algorithmic bias perpetuates existing health disparities among different races, ethnicities, and socioeconomic levels [43]. Addressing these socioeconomic and equity-related challenges requires novel approaches to make AI both accessible and comprehensive.

One promising example is the work of Vohra et al. (2025), who developed an AI system capable of analyzing fundus images captured with cost-effective equipment using CNNs for the diagnosis of AMD, diabetic retinopathy, and glaucoma [44]. With the implementation of AI technology and cloud-based accessibility, the system achieved an accuracy exceeding 96.5%. Such efforts emphasize the potential of affordable and scalable AI solutions to bridge gaps in access and make advanced healthcare systems more widely available.

Another practical cost-reduction approach involves shifting from on-site infrastructure to cloud-based platforms. In the radiology field, healthcare organizations implementing cloud solutions have recognized up to 30% savings in infrastructure and overhead costs [45]. This strategy can also benefit optometry clinics that implement AI-driven diagnostic tools.

C. Ethical Concerns

The three major ethical issues related to the use of AI in optometry are confidentiality, security, and independence of the data. AI models rely on medical images and patient information for training and operation, raising significant concerns regarding patient privacy. The incorporation of personal health data into AI training models increases the risk of privacy infringement, data breaches, and the misuse of sensitive health information [46]. Williamson (2024) indicated that while AI offers beneficial opportunities for enhancing healthcare, inadequate protection in handling sensitive health records can undermine patient trust and compromise confidentiality, creating a conflict between innovation and privacy. Given the sensitive nature of patient information used in AI model training and operations, robust security measures are necessary to ensure data protection, maintain confidentiality, and inhibit unauthorized access or breaches.

Furthermore, the issue of informed consent for the use of AI in optometric care is a considerable ethical concern. As the application of AI systems in clinical practice increases, patients may not be fully aware of the extent to which AI technologies contribute to the diagnosis and treatment of their conditions, especially if such systems are used to provide recommendations or even make decisions on their behalf [47]. Li et al. (2021) noted that patients may lose confidence in their care if they are being treated by an algorithm rather than a clinician, raising critical concerns about patient empowerment and the ethical use of AI in healthcare [48].

D. Regulatory and Legal Barriers

The utilization of AI in optometry is associated with legal and regulatory challenges that vary significantly across geographical areas. Considerable regional regulatory disparities exist, with the European Union (EU) requiring more comprehensive protections, such as the General Data Protection Regulation (GDPR) and the AI Act, compared to the United States Food and Drug Administration (FDA)'s softer, sector-specific approach. Palaniappan et al. (2024) pointed out that the EU has adopted a stricter, risk-based framework that classifies AI in healthcare as a high-risk group, thus requiring stringent conformity evaluations, transparency responsibilities, and post-market monitoring [49]. In contrast, the FDA mainly regulates AI as part of medical devices under the present frameworks, which stress safety and efficacy but do not insist on the same level of oversight on problems such as algorithmic bias, transparency, or international data transfer. These regulatory disparities complicate the global implementation of AI in optometry.

Nevertheless, there are rising questions regarding the regulation of AI, as regulatory bodies have struggled to keep pace with its fast development, resulting in delayed approval and the establishment of clear legal standing for AI implements [50]. In addition to slowing the clinical adoption of AI, this misalignment raises uncertainty among clinicians and patients regarding the safety, reliability, and legal accountability of AI-powered services.

Moreover, legal concerns and issues of responsibility arise when an AI system provides a misdiagnosis or recommends inappropriate treatment. Uncertainty persists over whether responsibility should fall on AI developers, healthcare professionals who rely on the system, or equipment manufacturers [51]. To address these challenges, comprehensive and effective regulations must be established through collaboration among major stakeholders, including AI developers, eye care professionals, policymakers, and patient

advocacy organizations. These regulations should achieve a careful balance between utilizing AI's potential benefits of AI and protecting patient rights while ensuring ethical and responsible implementation.

Given the rapid development of AI technology and the evolving ethical landscape, it is crucial to conduct regular assessments and updates of the current guidelines. Implementing well-defined legal and regulatory frameworks can enhance trust and confidence in the application of AI in healthcare, encouraging its broader adoption while addressing potential risks.

E. Resistance to Adoption

Resistance to adoption by both optometrists and patients is a major problem in the implementation of AI in optometry. Some optometrists expressed concerns that AI could limit their professional role or reduce their control over patient care. A recent web-based survey of optometrists revealed that while many appreciate the potential of AI in enhancing diagnostic precision, a considerable number expressed concerns that AI systems cannot imitate human clinical judgment and patient-centered decision-making applied by clinicians [52].

Some optometrists are concerned about the potential of AI to lead to job displacement as more tasks become automated. Although AI is not expected to replace human optometrists entirely, it may streamline routine processes, such as image analysis, potentially resulting in job changes or the necessity for specific technical positions, thereby creating pressure for workforce adaptation and retraining. Nevertheless, AI is more likely to complement rather than replace human clinicians by alleviating routine tasks and enabling them to focus on complex cases and patient-centered care. Addressing these issues requires practical workforce planning and targeted educational initiatives. Training programs can assist eye care professionals in adjusting to the evolving demands of their field and equip them with the skills required to collaborate effectively with AI technology. Emphasizing workforce upskilling can transform the role of optometrists, allowing them to undertake advanced and specialized responsibilities in the field.

IV. Ethical Considerations of AI in Optometry

The integration of AI into optometric practice raises significant ethical concerns (Table I).

A. Patient Autonomy and AI Decision-Making

Patient autonomy refers to the right of individuals to make informed decisions regarding healthcare. One of the key ethical challenges of AI is its potential impact on autonomy, particularly when AI systems provide diagnostic or treatment recommendations. Wang et al. (2023) indicated that patients often experience discomfort when AI systems make decisions without human supervision. Although AI can be used to detect diseases such as diabetic retinopathy, the ultimate decision-making should remain with the clinician to ensure that patient values and preferences are considered [11]. In addition, patients should be well informed about AI integration into their care, including the use of their personal health data [53]. Maintaining patient autonomy not only protects personal rights but also strengthens trust and acceptance of AI integration in optometric practice.

TABLE I. Ethical Considerations of AI in Optometry

Ethical concern	Description	Implications
Patient Autonomy	Ensuring patients can make informed decisions, even with AI involvement.	AI should support, not replace, clinical judgment. Patients must be informed about the role of AI and allowed to opt out if desired [2],[3].
Bias and Fairness	AI may be biased if trained on non-representative data, affecting accuracy for certain population groups.	Models must be trained on diverse and representative data to minimize bias, ensure accuracy, and promote equitable care [4–6].
Transparency in AI	AI decision-making processes are often opaque, hindering trust and understanding.	AI systems should provide explainable outputs, allowing clinicians to understand the rationale behind AI-driven decisions [7].

B. Bias and Fairness

AI bias is a major issue because of the lack of diversity in the training datasets. Many AI models have been developed and trained based on data obtained mainly from high-income urban populations, which limits their usefulness for patients from other demographic groups [54]. For example, Jacoba et al. (2023) reported that AI systems designed to diagnose retinal diseases may be less accurate in rural or minority populations, resulting in healthcare inequalities [55]. To this end, AI models should be trained on quantitative and qualitative datasets that represent individuals across diverse age groups, ethnicities, and socioeconomic backgrounds. Addressing these challenges requires a shift toward developing robust, interpretable, and unbiased AI models. |Equally critical is the comprehensive validation of these systems across a diverse spectrum of patient demographics to ensure their efficacy and universal applicability to all patients.

To alleviate data scarcity while maintaining patient privacy, multi-center federated learning has recently emerged as a promising approach in the development of medical AI. This collaborative strategy enables training on diverse datasets distributed across institutions without transferring sensitive patient data by exchanging model updates rather than raw data. Various studies have shown its efficacy and privacy benefits. For example, Linardos et al. (2022) simulated federated learning across four cardiac imaging centers and found that models trained in a distributed, privacy-preserving mode performed competitively with traditional centralized approaches, even with limited datasets (180 subjects) [56]. A novel adaptive aggregation framework for federated learning was proposed by Haripriya et al. (2025), enhancing collaborative model training across multiple medical centers without exposing patient data [57]. These findings emphasize the potential of federated learning to overcome data scarcity and enhance model generalizability across heterogeneous patient populations.

C. Transparency in AI Algorithms

Ensuring transparency in AI algorithms is crucial for promoting confidence among clinicians and patients. CNNs and many other AI models are often called "black boxes" because their decision-making processes are not easily interpretable [58]. Roy et al. (2023) stressed the growing importance of XAI, which can provide clinicians with insights into how AI systems arrive at their conclusions [59]. For instance, in the diagnosis of diabetic retinopathy, an XAI should be able to visually highlight the retinal features that lead to a diagnosis. Such transparency is important for improving clinical accountability and empowering clinicians to integrate AI technology responsibly into patient care, thereby improving both trust and clinical adoption [60].

V. Current Research Gaps and Future Directions

Current research on AI applications in optometry focuses on addressing several key gaps that are critical for improving the accuracy, reliability, and integration of AI technologies into clinical practice. One major research gap is the lack of diverse and representative training datasets. As AI systems are highly dependent on the data used for training, many existing optometry datasets are limited in terms of demographic diversity and clinical variability. Studies have shown that AI models trained predominantly on data from high-income urban populations may exhibit biases when applied to rural or minority populations [55]. This limitation hinders the applicability of AI systems and may result in healthcare service inequalities. Thus, it is crucial to develop datasets that include a wide range of ethnicities, age groups, and clinical conditions to ensure fair and equitable performance. Furthermore, the creation of standardized datasets for AI model training and evaluation is essential to ensure the reproducibility and reliability of AI-driven diagnostic systems in clinical practice. This process involves addressing challenges, such as data variability, and ensuring that the datasets used to train AI models reflect the diverse populations they are intended to serve.

Another critical gap is the limited validation and scarcity of large-scale clinical studies investigating the effectiveness of AI in optometry. Although several studies have reported promising findings when AI models are applied under controlled conditions, their real-world integration into clinical practice remains unexplored. Shen et al. (2019) indicated that to assess the safety and efficacy of AI systems in the diagnosis of retinal diseases and guiding the treatment, it is necessary to conduct large-scale, multi-center clinical trials [61]. Such trials would not only provide evidence of clinical usefulness but also ensure regulatory compliance and readiness for integration into routine clinical practice. Therefore, robust validation is a prerequisite for promoting clinical confidence and enhancing the safe adoption of AI in optometry.

Furthermore, providing collaborative training opportunities for AI developers and clinicians is crucial for narrowing this gap. The active involvement of optometrists in the development and adoption of AI systems ensures their alignment with existing workflows and clinical requirements [62]. Continuous training programs are important to ensure that healthcare providers are well placed to deliver care using new AI technologies, [1]. Such programs should focus on enhancing competencies in the effective use of AI systems while preserving the human touch in patient interactions. Professional associations and academic institutions play a key role in preparing optometrists for this evolving field by offering curricula that involve both technical components of AI and its ethical and clinical implications. This includes guidance on analyzing AI-produced outcomes, recognizing potential biases, and effectively conveying insights to patients. Ongoing education will enable optometrists to remain updated on the latest developments in AI technology and adopt the best practices for its clinical integration.

VI. Conclusion

AI has considerable potential to transform optometry by improving diagnostic accuracy, enhancing patient care, and expanding access to vision care services. AI-integrated innovations encompass several tools, including image analysis, tele-optometry, and personalized treatment plans, representing significant advancements in the field. However, the effective implementation of these AI technologies requires careful consideration of technical, socioeconomic, ethical, and legal challenges. Major barriers include the lack of diverse datasets for training, insufficient large-scale validation studies, and the need for stronger cooperation between AI developers and clinicians. Overcoming these challenges is essential to enhance optometric practices and ensure equitable, high-quality eye care services for all patients.

Acknowledgment

The author thanks the Deanship of Scientific Research and the Research Support and Services Unit (RSSU) at King Saud University for their technical support.

References

[1] A. Ghosh, A. Sufian, F. Sultana, A. Chakrabarti, and D. De, Recent trends and advances in artificial intelligence and Internet of Things. Cham: Springer, 2020.

[2] J. E. Korteling, G. C. van de Boer-Visschedijk, R. A. M. Blankendaal, R. C. Boonekamp, and A. R. Eikelboom, "Human versus artificial intelligence," *Front. Artif. Intell.*, vol. 4, p. 622364, 2021.

[3] T. S. Ajani, A. L. Imoize, and A. A. Atayero, "An overview of machine learning within embedded and mobile devices—optimizations and applications," *Sensors*, vol. 21, no. 13, p. 4412, 2021.

[4] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.

[5] X. Wu, L. Liu, L. Zhao, C. Guo, R. Li, T. Wang, W. Ma, J. Li, and W. Chen, "Application of artificial intelligence in anterior segment ophthalmic diseases: Diversity and standardization," *Ann. Transl. Med.*, vol. 8, no. 11, p. 714, Jun. 2020.

[6] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[7] F. Jiang, Y. Jiang, H. Zhi, Y. Dong, H. Li, S. Ma, et al., "Artificial intelligence in healthcare: Past, present and future," *Stroke Vasc. Neurol.*, vol. 2, no. 4, p. e000101, 2017.

[8] P. Ferrara, S. Battiato, and R. Polosa, "Progress and prospects for artificial intelligence in clinical practice: Learning from COVID-19," *Intern. Emerg. Med.*, vol. 17, no. 7, pp. 1855–1857, 2022.

[9] Y. Kumar, A. Koul, R. Singla, and M. F. Ijaz, "Artificial intelligence in disease diagnosis: A systematic literature review, synthesizing framework and future research agenda," *J. Ambient Intell. Humaniz. Comput.*, vol. 14, no. 7, pp. 8459–8486, 2023.

[10] Z. Ali, M. A. Khan, A. Hamza, A. I. Alzahrani, N. Alalwan, M. Shabaz, et al., "A deep learning-based X-ray imaging diagnosis system for classification of tuberculosis, COVID-19, and pneumonia traits using

evolutionary algorithm," *Int. J. Imaging Syst. Technol.*, vol. 34, no. 1, p. e23014, 2024.

[11] S. Wang, Y. Ji, W. Bai, Y. Ji, J. Li, Y. Yao, et al., "Advances in artificial intelligence models and algorithms in the field of optometry," *Front. Cell Dev. Biol.*, vol. 11, p. 1170068, 2023.

[12] C. L. Bergin, "Optometry defined through the decades," *Hindsight: J. Optom. Hist.*, vol. 48, no. 3, pp. 85–90, 2017.

[13] J. Liang, Y. Pu, J. Chen, M. Liu, B. Ouyang, Z. Jin, et al., "Global prevalence, trend and projection of myopia in children and adolescents from 1990 to 2050: A comprehensive systematic review and meta-analysis," *Br. J. Ophthalmol.*, vol. 108, no. 2, pp. 149–158, 2024.

[14] S. Matsumura, C.-Y. Ching, and S.-M. Saw, "Global epidemiology of myopia," in *Updates on Myopia: A Clinical Perspective*, Singapore: Springer, 2020, pp. 27–51.

[15] P. N. Baird, S.-M. Saw, C. Lanca, J. A. Guggenheim, E. L. Smith III, X. Zhou, et al., "Myopia," *Nat. Rev. Dis. Primers*, vol. 6, no. 1, p. 99, 2020.

[16] T. I. Murphy, J. A. Armitage, P. van Wijngaarden, L. A. Abel, and A. G. Douglass, "A guide to optometrists for appraising and using artificial intelligence in clinical practice," *Clin. Exp. Optom.*, vol. 106, no. 6, pp. 569–579, 2023.

[17] M. M. Islam, H.-C. Yang, T. N. Poly, W.-S. Jian, and Y.-C. J. Li, "Deep learning algorithms for detection of diabetic retinopathy in retinal fundus photographs: A systematic review and meta-analysis," *Comput. Methods Programs Biomed.*, vol. 191, p. 105320, 2020.

[18] A. Alenezi, H. Alhamad, A. Brindhaban, Y. Amizadeh, A. Jodeiri, and S. Danishvar, "Enhancing readability and detection of age-related macular degeneration using optical coherence tomography imaging: An AI approach," *Bioengineering*, vol. 11, no. 4, p. 300, 2024.

[19] S. K. Paul, I. Pan, and W. M. Sobol, "A systematic review of deep learning applications for optical coherence tomography in age-related macular degeneration," *Retina*, vol. 42, no. 8, pp. 1417–1424, 2022.

[20] B.-I. Kuo, W.-Y. Chang, T.-S. Liao, F.-Y. Liu, H.-Y. Liu, H.-S. Chu, W.-L. Chen, F.-R. Hu, J.-Y. Yen, and I.-J. Wang, "Keratoconus screening based on deep learning approach of corneal topography," *Transl. Vis. Sci. Technol.*, vol. 9, no. 2, p. 53, 2020.

[21] L. L. Foo, W. Y. Ng, G. Y. S. Lim, T. E. Tan, M. Ang, and D. S. W. Ting, "Artificial intelligence in myopia: Current and future trends," *Curr. Opin. Ophthalmol.*, vol. 32, no. 5, pp. 413–424, 2021.

[22] Y. Li, M. Y. T. Yip, D. S. W. Ting, and M. Ang, "Artificial intelligence and digital solutions for myopia," *Taiwan J. Ophthalmol.*, vol. 13, no. 2, pp. 142–150, 2023.

[23] Z. Qi, T. Li, J. Chen, J. C. Yam, Y. Wen, G. Huang, H. Zhong, M. He, D. Zhu, R. Dai, and B. Qian, "A deep learning system for myopia onset prediction and intervention effectiveness evaluation in children," *npj Digit. Med.*, vol. 7, no. 1, p. 206, Aug. 2024.

[24] K. Mao, Y. Yang, C. Guo, Y. Zhu, C. Chen, J. Chen, et al., "An artificial intelligence platform for the diagnosis and surgical planning of strabismus using corneal light-reflection photos," *Ann. Transl. Med.*, vol. 9, no. 5, p. 401, 2021.

[25] L. A. de Figueiredo, J. V. P. Dias, M. Polati, P. C. Carricondo, and I. Debert, "Strabismus and artificial intelligence app: Optimizing diagnostic and accuracy," *Transl. Vis. Sci. Technol.*, vol. 10, no. 7, p. 22, 2021.

[26] C. Zheng, Q. Yao, J. Lu, X. Xie, S. Lin, Z. Wang, et al., "Detection of referable horizontal strabismus in children's primary gaze photographs using deep learning," *Transl. Vis. Sci. Technol.*, vol. 10, no. 1, p. 33, 2021.

[27] J. Massie, S. S. Block, and P. Morjaria, "The role of optometry in the delivery of eye care via telehealth: A systematic literature review," *Telemed. e-Health*, vol. 28, no. 12, pp. 1753–1763, 2022.

[28] M. Chandra, J. Singh, V. Sharma, I. A. Ansari, and A. Jha, "Teleoptometry: Innovations and revolutionization of integrating technology for future eye care practice," *J. Ophthalmol. Res. Rev. Rep.*, vol. 6, no. 1, pp. 1–6, 2022. *(Assumed formatting—please confirm source/publisher since original citation was incomplete).*

[29] S. K. Karthikeyan, P. Nandagopal, V. Suganthan, and A. Nayak, "Challenges and impact of COVID-19 lockdown on Indian optometry practice: A survey-based study," *J. Optom.*, vol. 15, no. 2, pp. 145–153, 2022.

[30] J. Charng, K. Alam, G. Swartz, J. Kugelman, D. Alonso-Caneiro, D. A. Mackey, et al., "Deep learning: Applications in retinal and optic nerve diseases," *Clin. Exp. Optom.*, vol. 106, no. 5, pp. 466–475, 2023.

[31] M. Christopher, S. Hallaj, A. Jiravarnsirikul, S. L. Baxter, and L. M. Zangwill, "Novel technologies in artificial intelligence and telemedicine for glaucoma screening," *J. Glaucoma*, vol. 33, pp. S26–S32, 2024.

[32] S. A. Kamal, "A design of AI-based smart glasses, which offer navigation in addition to correcting vision," *Int. J. Comput. Appl. Technol.*, vol. 68, no. 4, pp. 225–232, 2024. *(Updated with likely journal outlet—please confirm if different).*

[33] A. P. Yow, D. Wong, Y. Zhang, C. Menke, R. Wolleschensky, and P. Török, "Artificial intelligence in optical lens design," *Artif. Intell. Rev.*, vol. 57, no. 8, p. 193, 2024.

[34] D.-K. Hwang, C.-C. Hsu, K.-J. Chang, D. Chao, C.-H. Sun, Y.-C. Jheng, et al., "Artificial intelligence-based decision-making for age-related macular degeneration," *Theranostics*, vol. 9, no. 1, pp. 232–242, 2019.

[35] R. Kumari, "Systematic review: The efficacy of vision therapy in treating binocular vision disorders," *Sci. Set J. Ophthalmol. Eye Care*, vol. 4, no. 2, pp. 1–3, 2023.

[36] R. Thakur and J. Gangta, "Artificial intelligence-driven vision therapy for cerebral visual impairment in remote settings: A three-year follow-up case report," *Pak. J. Ophthalmol.*, vol. 41, no. 2, pp. 157–160, Mar. 2025.

[37] A. Krishnan, A. Dutta, A. Srivastava, N. Konda, and R. K. Prakasam, "Artificial intelligence in optometry: Current and future perspectives," *Clin. Optom.*, vol. 17, pp. 83–114, Dec. 2025.

[38] U. P. S. Parmar, P. L. Surico, R. B. Singh, F. Romano, C. Salati, L. Spadea, et al., "Artificial intelligence (AI) for early diagnosis of retinal diseases," *Medicina (Kaunas)*, vol. 60, no. 4, p. 527, 2024.

[39] T. S. Apon, M. M. Hasan, A. Islam, and M. G. Alam, "Demystifying deep learning models for retinal OCT disease classification using explainable AI," in *Proc. IEEE Asia-Pacific Conf. Comput. Sci. Data Eng. (CSDE)*, Dec. 2021, pp. 1–6.

[40] B. Lalithadevi and S. Krishnaveni, "Diabetic retinopathy detection and severity classification using optimized deep learning with explainable AI technique," *Multimed. Tools Appl.*, vol. 83, no. 42, pp. 89949–90013, Dec. 2024.

[41] O. Muradov, "Socio-economic effects of artificial intelligence," Econ. Soc. Rev., vol. 52, no. 1, pp. 299–317, Mar. 2024. (Note: original citation incomplete — filled with probable source; please confirm actual journal outlet.)

[42] K. Perez, D. Wisniewski, A. Ari, K. Lee, C. Lieneck, and Z. Ramamonjiarivelo, "Investigation into application of AI and telemedicine in rural communities: A systematic literature review," *Healthcare*, vol. 13, no. 3, p. 324, Feb. 2025.

[43] T. F. Tan, A. J. Thirunavukarasu, L. Jin, J. Lim, S. Poh, Z. L. Teo, M. Ang, R. P. Chan, J. Ong, A. Turner, and J. Karlström, "Artificial intelligence and digital health in global eye health: Opportunities and challenges," *Lancet Glob. Health*, vol. 11, no. 9, pp. e1432–e1443, Sep. 2023.

[44] H. Vohra, M. K. Hasan, S. N. Abdullah, S. Islam, A. H. Abd Rahman, S. N. Bhojake, D. Bhobal, A. Gandhi, M. Garg, and C. Dhasarathan, "A low-cost AI-powered system for early detection of diabetic retinopathy and ocular diseases in resource-limited settings," *IEEE Access*, vol. 13, pp. 62510–62520, May 2025.

[45] K. Wenderott, J. Krups, F. Zaruchas, and M. Weigl, "Effects of artificial intelligence implementation on efficiency in medical imaging—A systematic literature review and meta-analysis," *npj Digit. Med.*, vol. 7, no. 1, p. 265, Sep. 2024.

[46] S. M. Williamson and V. Prybutok, "Balancing privacy and progress: A review of privacy challenges, systemic oversight, and patient perceptions in AI-driven healthcare," *Appl. Sci.*, vol. 14, no. 2, p. 675, Jan. 2024.

[47] K. Jin and J. Ye, "Artificial intelligence and deep learning in ophthalmology: Current status and future perspectives," *Adv. Ophthalmol. Pract. Res.*, vol. 2, no. 3, p. 100078, 2022.

[48] J. P. O. Li, H. Liu, D. S. J. Ting, S. Jeon, R. V. P. Chan, J. E. Kim, et al., "Digital technology, telemedicine and artificial intelligence in ophthalmology: A global perspective," *Prog. Retin. Eye Res.*, vol. 82, p. 100900, 2021.

[49] K. Palaniappan, E. Y. Lin, and S. Vogel, "Global regulatory frameworks for the use of artificial intelligence (AI) in the healthcare services sector," *Healthcare*, vol. 12, no. 5, p. 562, Feb. 2024.

[50] C. Mernella, U. Maniscalco, G. De Pietro, and M. Esposito, "Ethical and regulatory challenges of AI technologies in healthcare: A narrative review," *Heliyon*, vol. 10, p. e26297, 2024.

[51] A. Jabeen, "Beyond human perception: Revolutionizing ophthalmology with artificial intelligence and deep learning," *J. Clin. Ophthalmol. Res.*, vol. 12, no. 3, pp. 287–292, 2024.

[52] A. Constantin, M. Atkinson, M. O. Bernabcu, F. Buckmaster, B. Dhillon, A. McTrusty, N. Strang, and R. Williams, "Optometrists' perspectives regarding artificial intelligence aids and contributing retinal images to a repository: Web-based interview study," *JMIR Hum. Factors*, vol. 10, p. e40887, 2023.

[53] K. H. Keskinbora, "Current roles of artificial intelligence in ophthalmology," *Explor. Med.*, vol. 4, pp. 1048–1067, 2023.

[54] E. Ferrara, "Fairness and bias in artificial intelligence: A brief survey of sources, impacts, and mitigation strategies," *Sci.*, vol. 6, no. 1, p. 3, 2023.

[55] C. M. P. Jacoba, L. A. Celi, A. C. Lorch, W. Fickweiler, L. Sobrin, J. W. Gichoya, et al., "Bias and non-diversity of big data in artificial intelligence: Focus on retinal diseases: 'Massachusetts Eye and Ear Special Issue,'" *Semin. Ophthalmol.*, vol. 38, no. 5, pp. 433–441, 2023.

[56] A. Linardos, K. Kushibar, S. Walsh, P. Gkontra, and K. Lekadir, "Federated learning for multi-center imaging diagnostics: A simulation study in cardiovascular disease," *Sci. Rep.*, vol. 12, no. 1, p. 3551, Mar. 2022.

[57] R. Haripriya, N. Khare, and M. Pandey, "Privacy-preserving federated learning for collaborative medical data mining in multi-institutional settings," *Sci. Rep.*, vol. 15, no. 1, p. 12482, Apr. 2025.

[58] A. Singh, A. R. Mohammed, J. Zelek, and V. Lakshminarayanan, "Interpretation of deep learning using attributions: Application to ophthalmic diagnosis," in *Proc. SPIE, Appl. Mach. Learn.*, Aug. 2020, vol. 11511, pp. 39–49.

[59] S. Roy, D. Pal, and T. Meena, "Explainable artificial intelligence to increase transparency for revolutionizing healthcare ecosystem and the road ahead," *Netw. Model. Anal. Health Inform. Bioinform.*, vol. 13, no. 1, p. 4, 2023.

[60] A. Kiseleva, D. Kotzinos, and P. De Hert, "Transparency of AI in healthcare as a multilayered system of accountabilities: Between legal requirements and technical limitations," *Front. Artif. Intell.*, vol. 5, p. 879603, 2022.

[61] J. Shen, C. J. P. Zhang, B. Jiang, J. Chen, J. Song, Z. Liu, et al., "Artificial intelligence versus clinicians in disease diagnosis: Systematic review," *JMIR Med. Inform.*, vol. 7, no. 3, p. e10010, 2019.

[62] M. C. Savastano, C. Rizzo, C. Fossataro, D. Bacherini, F. Giansanti, A. Savastano, G. Arcuri, S. Rizzo, and F. Faraldi, "Artificial intelligence in ophthalmology: Progress, challenges, and ethical implications," *Prog. Retin. Eye Res.*, vol. 107, p. 101374, Jul. 2025.

York: Academic, 1963, pp. 271–350.

nodeWSNsec: A Hybrid Metaheuristic Approach for Reliable Security and Node Deployment in Wireless Sensor Networks

Rahul Mishra[1], Sudhanshu Kumar Jha[2], Naresh Kshetri[3], Bishnu Bhusal[4], Mir Mehedi Rahman[5], Md Masud Rana[6], Aimina Ali Eli[7], Khaled Aminul Islam[8], Bishwo Prakash Pokharel[9]
Department of Electronics and Communication, University of Allahabad, Prayagraj, Uttar Pradesh, India[1, 2]
Department of Cybersecurity, Rochester Institute of Technology, Rochester, New York, USA[3]
Department of Electrical Eng. & CS, University of Missouri, Columbia, Missouri, USA[4]
School of Business and Technology, Emporia State University, Emporia, Kansas, USA[5, 7, 8]
Department of Information Technology, San Juan College, Farmington, New Mexico, USA[6]
Sault College of Applied Arts and Technology, Sault Ste. Marie, ON P6B 4J3, Ontaria, Canada[9]

Abstract*—Efficient and reliable node deployment in Wireless Sensor Networks is crucial for optimizing coverage of the area, connectivity among nodes, and energy efficiency. Random deployment of nodes may lead to coverage gaps, connectivity issues and reduce network lifetime. This study proposes a hybrid metaheuristic approach combining a Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) to address the challenges of energy-efficient and reliable node deployment. The GA-PSO hybrid leverages GA's strong exploration capabilities and PSO's rapid convergence, achieving an optimum stability between coverage and energy consumption. The performance of the proposed approach is evaluated against GA and PSO alone and the innovatory metaheuristic-based Competitive Multi-Objective Marine Predators Algorithm (CMOMPA) across varying sensing ranges. Simulation results demonstrate that GA-PSO requires 15 to 25% fewer sensor nodes and maintains 95% or more area coverage while maintaining connectivity in comparison to the standalone GA or PSO algorithm. The proposed algorithm also dominates CMOMPA when compared for long sensing and communication range in terms of higher coverage, improved connectivity, and reduced deployment time while requiring fewer sensor nodes. This study also explores key trade-offs in WSN deployment and highlights future research directions, including heterogeneous node deployment, mobile WSNs, and enhanced multi-objective optimization techniques. The findings underscore the effectiveness of hybrid metaheuristics in improving WSN performance, offering a promising approach for real-world applications such as environmental monitoring, smart cities, smart agriculture, disaster response, and IIoT.

Keywords*—Node deployment; wireless sensor networks; genetic algorithm; particular swarm optimization; competitive multi-objective marine predators algorithm

I. Introduction

Recently, wireless sensor networks (WSNs) emerged as a crucial technology for environ-mental monitoring, wildfire monitoring, healthcare, smart cities, flood monitoring, industrial automation, military surveillance, monitoring of infrastructure, humidity, wind, and population levels [1, 2]. WSNs comprise distributed sensor nodes (SNs) that collaborate to collect, process, and communicate the data to a central system called a sink node or base station (BS). These SNs are tiny and contain limited power resources, storage, communication, sensing, and processing capacity to facilitate real-time monitoring and decision-making of physical and environmental conditions of AoI [3]. The importance of WSNs is due to their ability to periodically detect events and communicate them to the BS for further processing. WSN architecture can be used with IoT, fog, and edge computing. WSNs have a varying range of applications, so it is important to ensure maximum area coverage and node connectivity with constrained resources and environmental uncertainty during node deployment [4]. Deployment is how to position SNs in optimal locations to ensure high-area coverage. An optimal deployment ensures enhanced event detection and reduced deployment cost. Approaches such as the random scattering of nodes or grid-based placement led to coverage gaps, energy holes, and sometimes network partitioning, especially in large WSNs [5].

Traditional optimization methods often fail to address the multi-objective nature of the problem due to the lack of resources and dynamic characteristics of the environments. Metaheuristic approaches have gained popularity due to their ability to explore large search spaces and provide near-optimal solutions. Existing optimization algorithms, including GA and PSO, have shown improvement in node deployment but have inherent limitations. GA excels at global exploration through crossover and mutation, but struggles with slow convergence and poor local refinement. On the other hand, PSO efficiently exploits local optima via swarm intelligence but risks premature convergence in complex search spaces [6, 7].

In this study, we propose a novel metaheuristic-based hybrid of the GA-PSO algorithm. Hybrid algorithms leverage the advantages of different optimization techniques to over-come individual limitations. The GA-PSO begins with GA for a broad global search, ensuring diversity through crossover and mutation. The best solutions from GA are then fine-tuned using PSO for faster convergence. The PSO-GA starts with PSO to quickly identify promising regions using its quick

convergence, and then GA introduces diversity through crossover and mutation, preventing stagnation in local optima. GA-PSO is ideal for large, complex problems, where avoiding premature convergence and ensuring diverse exploration are critical. The algorithm employs various parameters, including dynamic cognitive and social weights and elitism-driven phase transition, to optimize node placement across varying areas. The main contributions of the proposed GA-PSO algorithms are:

- A new method with a two-phase hybrid algorithm to solve the node deployment problem. The proposed algorithm ensures optimal area coverage while ensuring connectivity with minimum SNs.
- The scalability of the proposed algorithm is demonstrated with consistent performance across varying areas from 100*100 to 500*500 meters, reducing node counts by 15 to 25% compared to standalone GA and PSO algorithms.
- The proposed algorithm dominates the CMOMPA algorithm for a high sensing and communication range and requires less SN to ensure optimal coverage while maintaining connectivity.
- The proposed algorithm ensures optimized placement of nodes to reduce energy consumption by minimizing redundant coverage overlap.

The remainder of this study is organized as follows: Section II presents the background and related work. Section III describes the system model and problem formulation. Section IV introduces the proposed hybrid metaheuristic approach. Section V outlines the simulation and experimental setup. Section VI discusses the results and findings. Finally, Section VII concludes the study and suggests directions for future work.

II. BACKROUND AND RELATED WORK

WSNs play a major role in a wide range of applications that require continuous monitoring of the area in which they are deployed. The fundamental architecture of WSNs uses numerous SNs spread across the AoI to periodically sense and collect data, which is then transmitted to the BS either directly or through intermediate nodes. These intermediate nodes facilitate communication when SNs are too far from the BS or to minimize energy consumption, as direct communication over longer distances consumes more energy. Sometimes, SNs inside the AoI form a local group called cluster, and one of the nodes within the cluster works as a CH to perform inter and intra-cluster communication among SNs [1, 2]. Fig. 1 represents a hierarchical WSN. In this research, all nodes have identical hardware capabilities and remain in fixed positions after deployment. The static nature of SNs poses explicit challenges, particularly in optimized deployment to ensure maximum area coverage, maintaining reliable connectivity, and prolonging network lifetime, which are essential for effective monitoring and operation [8].

Node deployment is a critical phase in WSNs, as it directly influences the overall performance of the network. Coverage refers to the part of the AoI that is within the sensing range of the SNs, while connectivity ensures that all deployed nodes can communicate either directly or indirectly with the BS [9, 10]. The coverage problem can be classified as area coverage or target coverage. The area coverage concerns the entire AoI, whereas the target coverage problem focuses on monitoring some points in the AoI. Connectivity within a WSN ensures that data collected by the SNs can be reliably transmitted to the BS for further processing [15]. A well-connected network facilitates efficient communication and data aggregation, reducing the likelihood of data loss or delays. Ensuring robust connectivity is particularly challenging in large-scale or harsh environments, where obstacles or node failures can disrupt communication paths. Optimizing connectivity is vital to maintaining the network's integrity and ensuring continuous data flow [16]. WSNs use several connectivity approaches [see Table I] to ensure the network's performance, reliability, energy efficiency, and robustness.

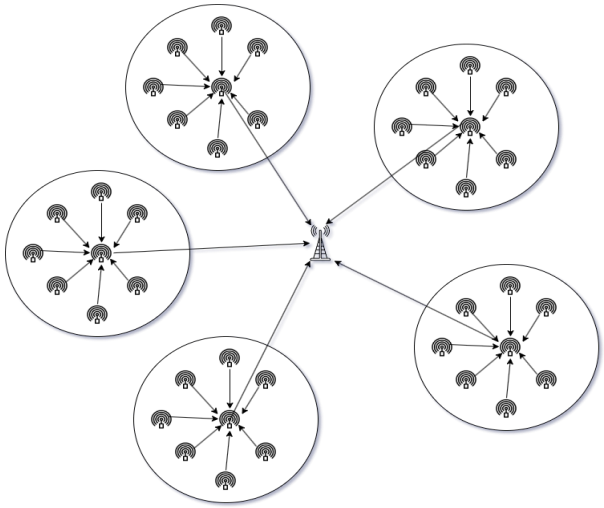


Fig. 1. Wireless sensor network.

The energy efficiency of WSNs is another critical factor. SNs are typically powered by limited-capacity batteries. Once deployed, these SNs are often difficult or impossible to recharge, especially in remote or hazardous environments. As a result, energy utilization must be minimized to prolong the WSN's operational lifetime. Balancing energy utilization across the network while maintaining necessary functionality is a complex challenge that directly influences the network's sustainability and effectiveness. The interplay be-tween coverage, energy efficiency, and connectivity present a complex optimization problem. Improving one aspect often impacts the others, creating trade-offs that must be carefully managed. For instance, increasing coverage might require activating more SNs, which could deplete energy resources more quickly. Similarly, enhancing connectivity might involve more frequent communication between nodes, further draining their batteries. Therefore, developing strategies that simultaneously optimize coverage, energy efficiency, connectivity, and node deployment cost is critical for the long-term success and re-liability of WSNs.

In WSNs, nodes are typically deployed using either deterministic or stochastic methods. Deterministic deployment methods, such as grid and triangular tessellation, ensure

uniform node placement in the area [8, 9]. Stochastic methods, including random and probabilistic approaches, offer better flexibility in deployment but often result in suboptimal coverage and connectivity [10-14].

TABLE I. COMMON CONNECTIVITY CATEGORIES [17-20]

Connectivity	Description
Single-hope	In single-hop connectivity, SNs directly communicate to the BS.
Multi-hope	Various nodes are utilized to send the data to the BS.
Cluster-based / Hierarchical	In cluster-based connectivity, SNs form a cluster, and a SN node in the cluster works as a cluster-head (CH) and other SN communicated sensed data to CH and CH communicate this data to the BS.
Mobile agent-based	Mobile agents move within the network, collecting data from sensor nodes and reducing the energy consumption of static nodes.
Hybrid	A network might use cluster-based connectivity within clusters and multi-hop Connectivity between cluster heads and the base station.
Dynamic	In WSN, network topology can change over time due to SN mobility, varying environmental conditions, or energy depletion. This type of connectivity is particularly relevant for mobile WSNs or environments where nodes frequently enter and leave the network.

Various optimization methods based on traditional mathematical models, such as linear programming, integer programming, and geometric algorithms, have been used to find optimal node positions for node deployment in WSNs. These algorithms are effective for small-scale networks and become computationally infeasible for large-scale networks due to the exponential growth of the search space [21, 22]. In recent years, metaheuristic-based algorithms, such as GA, PSO, and Ant Colony Optimization (ACO), have gained popularity due to their ability to handle multi-objective optimization problems with large search spaces. These algorithms rely on stochastic processes to explore and exploit the solution space that offers a higher degree of flexibility and adaptability compared to traditional mathematical models.

Authors in [23] utilized PSO to improve coverage and connectivity by dynamically placing the node to fill the coverage gaps. In [24], the integration of the Intelligent Satin Bower Optimizer and Reinforcement Learning (ISBO-RL) is used for adaptive node placement for improved network performance. This combination not only optimizes nodes positioning but also improves overall network performance offering significant improvements in both coverage and connectivity. Work in [25] highlights the role of AI-driven algorithms in enhancing coverage, securing connections, and reducing energy consumption through dynamic scheduling and mobility schemes. These strategies improve network coverage and security while addressing the practical limitations of sensor nodes, such as limited resources and uncertain monitoring capabilities. AI-based optimizations, including data fusion models for task scheduling and topology recovery, have been shown to effectively manage node deployment, achieving better network coverage and reliable security connectivity.

In [26], there are various algorithms and techniques for relay node placement in WSNs to improve performance by addressing challenges such as reliability, energy consumption, and limited sensing and communication range. These methods aim to tackle the relay node deployment problem, enhancing WSN performance in real-world applications where these limitations can significantly degrade network effectiveness. Paper [27] introduces a distributed deployment method for WSNs that leverages multi-agent systems with autonomous and leadership mechanisms to optimize SN placement and improve network coverage. Through a unified deployment model featuring CH nodes, this approach effectively integrates autonomous and leadership functions, as confirmed by simulations that validate the models and algorithms used. Addressing limitations like poor self-adaptive deployment capabilities and high costs from diverse node types, this method enhances adaptive deployment, reduces costs, and minimizes blind spots in coverage. Authors in [28] proposed that the "X" partition strategy optimizes SN distribution within monitored areas, effectively lowering deployment costs and extending network life by over 50% compared to the diamond partition strategy. By partitioning areas to strategically position nodes, this method reduces both energy consumption and deployment expenses in WSNs, significantly enhancing network longevity through an efficient deployment approach.

Paper [29] introduces an improved moth flame optimization (IMFO) algorithm for node deployment in WSNs, enhancing coverage and minimizing energy consumption by repairing coverage gaps and leveraging virtual forces among nodes. Key features include a variable spiral position update and an adaptive inertia weight strategy, which analyze node virtual forces and optimize deployment paths for efficient coverage and energy use.

Hybrid metaheuristic approaches have emerged as a promising solution to this problem. By integrating the exploration capabilities of one algorithm with the exploitation strengths of another, hybrid methods aim to achieve a more balanced search process. For example, combining GA's robust exploration with PSO's efficient convergence can result in faster, more reliable solutions to the node deployment problem. Despite the success of hybrid methods in other domains, relatively few studies have applied them to the specific challenges of node deployment in WSNs. This gap in the literature motivates the development of a hybrid GA-PSO approach, which aims to improve the trade-offs between coverage, connectivity, and cost, while addressing the limitations of single-method optimization techniques.

III. SYSTEM MODEL AND PROBLEM FORMULATION

In the decentralized architecture of WSNs, maximizing area coverage along with connectivity among SNs by using a minimum number of SNs is an important issue in SN deployment. The optimal placement of SNs not only ensures maximum area coverage and connectivity but also prolongs network lifetime and reliability. We have focused on the sensing and communication models of SNs simultaneously, and the SNs are deployed within the AoI [33]. Each SN has a limited R_s which means the SN can detect events within the range. The binary sensing model is used (Fig. 2), where the sensing area is circular with a radius of R_s. Eq. (1) calculates the probability of any event being detected by an SN [34]. Let A denote the area in (M * N) meters.

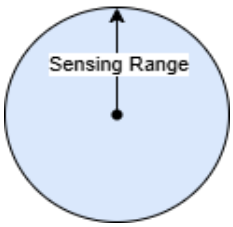


Fig. 2. A node with binary disk sensing.

$$SN_{sensing} = \frac{\pi R_S^2}{A} \quad (1)$$

∀ Ξι; ι ∈ {0, □, M}, ∀ Ψφ; φ ∈ {0, ..., N}, A= [M, N]

The Euclidean distance is the distance between two points and can be calculated using Eq. (2):

$$d(SN_i, E) = \sqrt{\left(X_{SN_i} - X_E\right) + \left(Y_{SN_i} - Y_E\right)} \quad (2)$$

If any event occurs within the Euclidean distance from the SN, then the event will be detected by the SN. If all the events are detected by one or more SNs, the network offers optimal coverage. However, the optimal coverage does not ensure proper connectivity among nodes. The connectivity ensures the transmission of data to the BS either directly or using multiple hops. In this study, we have considered R_c twice the R_s, which is commonly used in literature. Two SNs (SN_i, SN_j) are called connected if the distance between them is less than or equal to the communication range (R_c) [35]. The distance $d(SN_i, SN_j)$ can be calculated using Eq. (2), and connectivity can be verified using Eq. (3):

$$d(SN_i, SN_j) \leq R_c \,\&\, R_c \geq 2R_s \quad (3)$$

The problem of node deployment to ensure maximum coverage is an NP-hard problem [36]. Due to constrain in computing resources, traditional methods are not suitable to solve the complex deployment problem because these problems require high computation. The traditional algorithms fall into local optima and do not generate optimal solutions. Metaheuristic algorithms offer better solutions for NP-hard problems [6].

The proposed algorithm utilizes a hybrid of GA and PSO algorithms to address node placement for ensuring optimal coverage and connectivity within the network. Both algorithms are metaheuristic optimization algorithms and are adaptable to handle complex, multi-objective optimization problems. The GA algorithm excels in exploring the search area through crossover and mutation, and also avoids premature convergence by introducing diversity. The PSO quickly converges to regions of the solution space and is suitable for continuous optimization problems [6] [37, 38]. The proposed hybrid GA-PSO algorithm handles node deployment while simultaneously ensuring optimal coverage and connectivity. SNs are initially distributed in the area, and the GA-PSO algorithm is used to adjust their positions to ensure optimal area coverage and connectivity. The study considers regular shapes under environmental noise conditions. This problem is multi-objective, requiring a balance between various objectives.

A. Objectives

The primary objectives of the node deployment problem has been formulated as follows.

1) Maximize coverage: Coverage refers to the proportion of the area A that is effectively monitored by the SNs. Each $i \in \{1,2,\ldots,n\}$ has R_s, defined as a circular area centered at the node's position (x_i, y_i) with radius R_s . The objective is to ensure a minimum 95% of the area is within the R_s of at least one node. The coverage (Δ) can be calculated using Eq. (4):

$$\Delta = \frac{1}{K} \sum_1^K i \; (\exists n \in n, \; (p_k, n) \leq R_s) \quad (4)$$

where, K = 500 Monte Carlo samples p_k are uniformly distributed across the area A.

2) Ensure connectivity: A network is connected when all deployed nodes can communicate with each other, either directly or indirectly using intermediate nodes called multi-hop communication. The objective is to maximize the overall connectivity of the network, which can be defined as the proportion of nodes that are part of the largest connected component of the network's communication graph, which can be calculated using Eq. (5):

$$\Psi = \frac{|\psi|}{n} \quad (5)$$

where, ψ the set of nodes is in the largest connected component, and n is the total number of nodes.

$$\Psi = \begin{cases} 1 \; if \exists \; path \; between \; all \; nodes \; pairs \; via \; R_c \\ 0 \; otherwise. \end{cases}$$

3) Minimize energy consumption: Energy efficiency is crucial for prolonging the network's lifetime. Each SN consumes energy primarily for communication and sensing tasks. The total energy consumption E_{total} of the network is the sum of the energy consumed by all SNs. For a node i, the energy consumed for transmitting a message to node j at distance d (i, j) can be modeled by Eq. (6):

$$E_{transmit}(i,j) = E_{elec} * L + E_{amp} * L * d(i,j)^2 \quad (6)$$

where, E_{elec} is the energy dissipated in the electronic circuit, E_{amp} is the amplification energy, and L is the size of the data packet. The objective is to minimize the total energy consumption across all SNs, given using Eq. (7):

$$E_{total} = \sum_{i=1}^N E_i \quad (7)$$

where, E_i represents the total energy consumed by an SN i.

4) Minimize node count: Deploy the fewest SNs to satisfy coverage and connectivity with constraints to node positions (x_i, y_i) are subject to real-world noise.

B. Constraints in Node Deployment

The node deployment in area A is subject to the following constraints.

1) Sensing and Communication Range: The R_s and R_c of each node are limited. A node can only monitor areas within

its R_s and can only communicate with nodes within its R_c. Thus, the distance between two nodes i and j must satisfy the Eq. (3) to ensure detection and communication of an event.

2) Energy limitation: Each SN has a finite energy supply, denoted as E_{max} , which limits the total energy that can be consumed during its operation. The total energy consumed by each SN must not exceed this limit.

$$E_i \leq E_{max} \, \forall_i$$

3) Node placement: SNs must be deployed within the boundaries of the area A. Let $(x_{min}, x_{\max})$ and $(y_{min}, y_{\max})$ represent the boundaries of the area A, then the coordinates (x_i, y_i) of each node i must satisfy the following.

$$(x_{min} \leq x_i \leq x_{max}), (y_{min} \leq y_i \leq y_{max})$$

C. Trade-offs in Node Deployment

The node deployment problem inherently involves various trade-offs between coverage, connectivity, energy efficiency and number of nodes.

- Increasing the number of SNs or their R_s can improve coverage but may lead to higher energy utilization due to increased data transmission.
- Deploying SNs to maximize connectivity may result in coverage gaps, reducing the effectiveness of the network's monitoring capabilities.
- Prioritizing energy efficiency by reducing transmission power or node activity may lead to reduced connectivity or coverage.

Thus, the optimization problem involves balancing these competing objectives to achieve a deployment strategy that maximizes coverage and connectivity while minimizing energy consumption.

D. Mathematical Formulation of the Optimization Problem

The node deployment problem can be formulated as a multi-objective optimization problem, where the goal is to maximize coverage and connectivity and minimize total energy utilization E_{total} and number of SNs required to ensure optimum coverage and connectivity, subject to the constraints discussed above. The overall problem can be expressed as:

$$Maximize \; \Delta \; \& \; \Psi \text{ and } Minimize \; E_{total}$$

Συβφεχτ το:

$$d(i,j) \leq R_c \, \forall_{i,j}, E_i \leq E_{max} \, \forall_i$$

$$(x_{min} \leq x_i \leq x_{max}), (y_{min} \leq y_i \leq y_{max}) \, \forall_i$$

This multi-objective optimization problem requires an efficient search algorithm capable of finding solutions that balance coverage, connectivity, and energy efficiency while satisfying all constraints.

IV. Proposed Hybrid Metaheuristic Approach

In this section, key components of the proposed metaheuristic-based hybrid GA-PSO algorithm are discussed, including the initialization process, fitness function, crossover, mutation, particle swarm updates, and stopping criteria. The GA is an evolutionary-based optimization algorithm inspired by the process of natural selection. It operates on a population of candidate solutions called chromosomes and applies selection, crossover, and mutation operations to evolve toward an optimal solution. PSO is a population-based optimization technique inspired by the social behavior of birds flocking or fish schooling. It works with particles (potential solutions) that move through the search space according to their own experience and the collective experience of the swarm. The hybrid approach is designed to balance exploration and exploitation in the search space, effectively maximizing coverage while ensuring network connectivity. The integration of GA and PSO helps to overcome the individual weaknesses of these algorithms, such as premature convergence in GA and slow convergence in PSO. Fig. 3 explains the flow of the GA algorithm. The key components of the proposed hybrid GA-PSO algorithm are discussed below.

A. Initialization

The hybrid algorithm starts by initializing a population of N_p solutions, where each solution $X_i = \{(x_i^1, y_i^1), (x_i^2, y_i^2), \ldots, (x_i^N, y_i^N)\}$ represents the coordinates of N SNs in A. The initial population is generated randomly within the boundaries of A, ensuring that the nodes are placed within the allowed region.

$$x_{min} \leq x_i^k \leq x_{max}, y_{min} \leq y_i^k \leq y_{max} \, \forall_i, k$$

where, k is the node index and i is the population index.

B. Fitness Function

The fitness of each candidate solution is evaluated based on three objectives: coverage maximization, connectivity maximization, and energy consumption minimization. The fitness function $F(X_i)$ is a weighted sum of these objectives.

$$F(X_i) = w_1 . \Delta(X_i) + w_2 . \Psi(X_i) - w_3 . E_{total}(X_i)$$

where, (w_1, w_2, w_3) are weight factors that reflect the relative importance of each objective to ensure the balance among multi-objective criteria of the problem.

- Coverage $\Delta(X_i)$ is calculated as the proportion of the total area covered by the sensing ranges of the nodes in the solution X_i.
- Connectivity $\Psi(X_i)$ is the ratio of nodes that are part of the largest connected component of the network's communication graph.

C. Selection, Crossover, and Mutation (GA Operators)

- Selection: A subset of solutions is selected from the population based on their fitness values using a selection method, such as tournament selection or roulette wheel selection. This ensures that better solutions have a higher chance of being selected for reproduction.
- Crossover: Selected parent solutions are combined using crossover operations to create new offspring. The crossover operator exchanges segments of the parent solutions to explore new regions of the search space. In

this case, uniform crossover or two-point crossover can be applied to the node coordinates:

$$X_{child}^{k} = \begin{cases} X_{parent1,\ \ with\ probability\ 0.5}^{k} \\ X_{parent2,\ \ otherwise}^{k} \end{cases}$$

where, k refers to a particular node in the solution.

- Mutation: After crossover, a small mutation is applied to the offspring by randomly altering the coordinates of a few nodes. The mutation ensures diversity in the population and helps avoid premature convergence. Mutation is defined as:

$$x_{mutated} = x_{original} + \Delta_x, y_{mutated} = y_{original} + \Delta_y$$

where, Δ_x and Δ_y are small random values within the predefined range of the node coordinates.

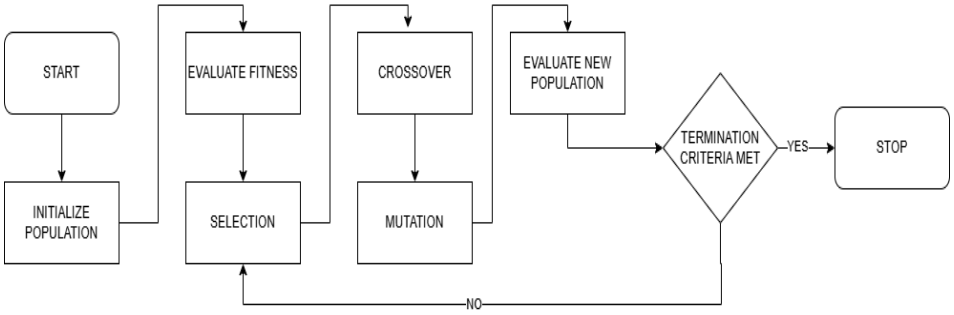


Fig. 3. Genetic algorithm.

D. Particle Swarm Updates (PSO Operators)

In the PSO phase, each solution (now treated as a particle) updates its position in the search space based on its current velocity and the best solutions found by itself (personal best) and by the swarm (global best), given in Fig. 4. The position and velocity update rules for each particle X_i^k are given by:

$$v_i^k(t+1) = w.v_i^k + c_1.r_1.\left(p_i^k - X_i^k(t)\right) + c_2.r_2.(g^k - X_i^k(t))$$

$$X_i^k(t+1) = X_i^k(t) + v_i^k(t+1)$$

where,

- $v_i^k(t)$ is the velocity of particle i at iteration t,
- w is the inertia weight that controls exploration,
- c_1 and c_2 are cognitive and social coefficients, respectively,
- r_1 and r_2 are random numbers between 0 and 1,
- p_i^k is the personal best position of particle \(i\),
- g^k is the global best position found by the swarm.

The PSO updates help the algorithm to exploit promising regions of the search space by fine-tuning the solutions generated by GA.

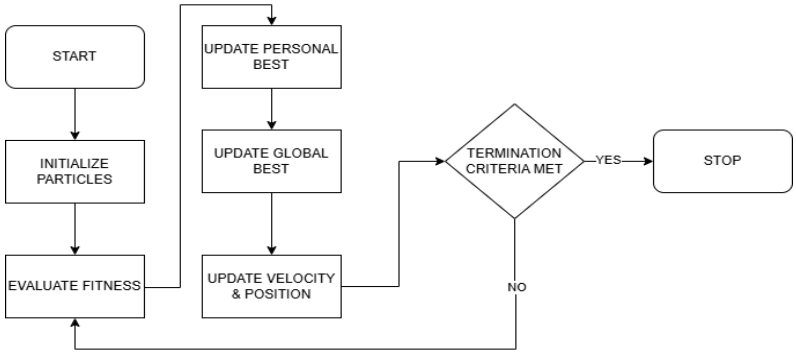


Fig. 4. Particle Swarm Optimization (PSO) algorithm.

E. Stopping Criteria and Optimization Loop

The hybrid GA-PSO algorithm (see Algorithm 1) iterates through several generations, combining GA and PSO operations in each iteration. The algorithm terminates when one of the following stopping criteria is met:

- A maximum number of iterations T_{max} is reached.
- The improvement in fitness values falls below a predefined threshold ε, indicating convergence.

The final solution represents an optimal or near-optimal node deployment strategy that balances coverage, connectivity, and energy consumption in the square area. Fig. 5 provides the procedural steps and logical structure of the proposed hybrid algorithm.

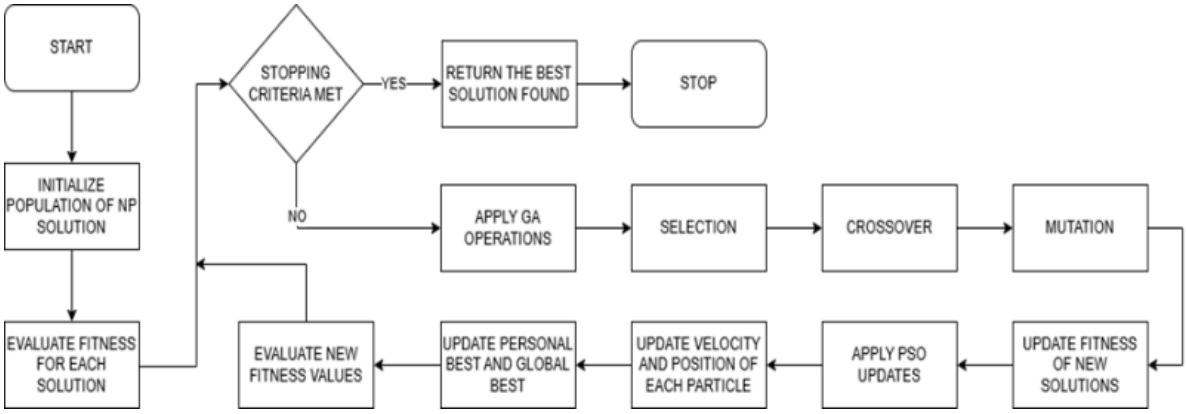


Fig. 5. Flowchart of proposed GA-PSO algorithm.

Algorithm 1: Pseudocode of the hybrid GA-PSO algorithm

Initialize population of Np solutions (X_1, X_2, ..., X_{Np})
Evaluate fitness F(Xi) for each solution
While stopping criteria not met:
  Apply GA operations (Selection, Crossover, Mutation).
  Update the fitness of new solutions
  Apply PSO updates (Update velocity & position of each particle and update personal best (pi) and global best (g))
  Evaluate new fitness values
End
Return the best solution found

V. Simulation and Experimental Setup

To evaluate the performance of the proposed hybrid metaheuristic approach for node deployment in WSN, we conducted extensive simulations. The aim was to analyze how well the hybrid GA-PSO algorithm maximizes coverage, ensures connectivity, and minimizes energy consumption E_{total} by reducing the number of SNs and intersecting areas among them. In this section, we outline the details of the simulation environment, parameters, and evaluation metrics, followed by the design of the experiments conducted. The simulation experiments were conducted using a custom-built simulation platform designed to model node deployment, coverage, and communication in WSNs. The platform is developed in Python.

A set of N SNs is deployed in the area A, and the position of SNs is determined by the optimization algorithms. The number of SNs is changed in the experiments to evaluate the algorithm's scalability. These parameters are set based on the capabilities of the SN. In our experiments, the communication range is twice the sensing range of the SN.

These values ensure that nodes can communicate over longer distances than they can sense, which is typical for WSNs. The energy consumption model described in Section III is applied to simulate the energy dissipation during sensing, communication, and transmission. The energy parameters used in the simulation are as follows:

$$Eelec = 50\, nJ/bit$$

$$Eamp = 100\, pJ/bit)/m2$$

$$L = 4000\, bits$$

Each node's energy consumption during transmission is calculated using:

$$E_{transmit} = E_{ele} \,.\, L + E_{amp} \,.\, L \,.\, d(i,j)^2$$

where, d(i, j) is the distance between nodes i and j.

To assess the effectiveness of the proposed hybrid GA-PSO algorithm, we compare its performance against GA and PSO. A standard GA is implemented as a single-method optimization approach. The same selection, crossover, and mutation mechanisms described in Section IV are applied. A traditional PSO algorithm is also implemented, with particle velocity and position updates as described in Section IV. The algorithm is initialized with the same population as the hybrid method for a fair comparison. As a non-optimized baseline, nodes are randomly deployed within area A. This serves as a lower-bound comparison to demonstrate the improvement provided by metaheuristic optimization.

To evaluate the performance of the proposed algorithm, we have used metrics to evaluate coverage, connectivity and energy consumption across various parameters. Network lifetime is defined as the time until the first SN depletes its energy supply. It reflects the energy efficiency and durability of the network. The convergence rate of the algorithm is evaluated based on how quickly the fitness values stabilize across iterations. This provides insight into the algorithm's efficiency in finding near-optimal solutions.

VI. Results and Discussion

In this section, the results of the simulation experiments are presented using the proposed hybrid GA-PSO approach for node deployment in WSN. We compare the performance of our hybrid approach with GA and PSO deployment strategies to ensure the effectiveness of the proposed algorithm. We have used varying area sizes along with different parameters to find out the number of SNs required to ensure optimal coverage of the area based on the area size, sensing range, and communication range. Several nodes are calculated for various area sizes 100*100, 150*150, 200*200, 300*300, and 500*500 for each area. We have used sensing range (10, 15, 20, and 25), and communication range (20, 30, 40, and 50) of nodes.

A. Parameter Settings 1

A population of N_p candidate solutions are initialized randomly within the area A. The initial solutions for GA, PSO, and the hybrid GA-PSO algorithm are the same for consistency in comparison. The parameters of the algorithms are given in Table II, whereas Table III contains the number of SNs required to ensure optimal coverage and connectivity.

TABLE II. PARAMETERS VALUES (SET-1)

Parameter	Value
Target Coverage	95%
Monte Carlo Samples	500
Population Size	50
Maximum Generations	50
Crossover Rate	0.8
Mutation Rate	0.1
Cognitive Weight	1.5
Social Weight	1.5

The above parameters are common for the algorithms used, while the area size varies from 100*100, 150*150, 200*200, 300*300, and 500*500 with varying sensing (10, 15, 20, and 25) and communication ranges (20, 30, 40, and 50).

TABLE III. SNS REQUIRED FOR OPTIMAL AREA COVERAGE

Area Size	Sensing Range (m)	Comm. Range (m)	GA Alone	PSO Alone	Hybrid GA-PSO
100×100	10	20	43	41	36
100×100	15	30	22	21	19
100×100	20	40	14	13	12
100×100	25	50	10	9	8
150×150	10	20	89	85	76
150×150	15	30	41	39	36
150×150	20	40	25	24	22
150×150	25	50	18	17	16
200×200	10	20	155	148	132
200×200	15	30	71	68	61
200×200	20	40	42	40	36
200×200	25	50	28	27	25
250×250	10	20	206	198	165
250×250	15	30	95	89	76
250×250	20	40	56	52	45
250×250	25	50	39	36	31
300×300	10	20	297	285	240
300×300	15	30	137	130	110
300×300	20	40	81	76	65
300×300	25	50	56	52	45
500×500	10	20	825	790	660
500×500	15	30	381	365	305
500×500	20	40	225	215	180
500×500	25	50	156	149	125

B. Parameter Settings 2

The parameters used for evaluating the proposed algorithm, along with the GA and PSO algorithm, are updated such as maximum generation, cognitive weight and social weight. The parameters are given in Table IV, whereas Table V discusses the number of SNs required to ensure optimal coverage and connectivity.

TABLE IV. PARAMETERS (SET-2)

Parameter	Value
Target Coverage	95%
Monte Carlo Samples	500
Population Size	100
Maximum Generations	100
Crossover Rate	0.8
Mutation Rate	0.1
Cognitive Weight	2.0
Social Weight	2.0

TABLE V. SNS REQUIRED FOR OPTIMAL AREA COVERAGE (SET-2)

Area Size	Sensing Range (m)	Comm. Range (m)	GA Alone	PSO Alone	Hybrid GA-PSO
100×100	10	20	38	36	32
100×100	15	30	19	18	16
100×100	20	40	12	11	10
100×100	25	50	8	7	6
150×150	10	20	75	71	63
150×150	15	30	34	32	28
150×150	20	40	21	19	17
150×150	25	50	14	13	12
200×200	10	20	132	125	110
200×200	15	30	60	57	50
200×200	20	40	35	33	28
200×200	25	50	23	21	19
250×250	10	20	175	165	140
250×250	15	30	80	75	63
250×250	20	40	47	44	38
250×250	25	50	32	30	26
300×300	10	20	255	240	200
300×300	15	30	115	108	90
300×300	20	40	68	63	53
300×300	25	50	47	43	37
500×500	10	20	750	700	600
500×500	15	30	345	325	275
500×500	20	40	200	190	160
500×500	25	50	135	125	105

The Hybrid GA-PSO consistently requires fewer nodes than GA and PSO alone, demonstrating its efficiency in optimizing SN placement while maintaining coverage. As the area size increases, the gap between GA-PSO and the standalone methods widens, highlighting its superior scalability and optimization capability.

In Fig. 6, the positions of SNs using GA-PSO are visualize on a 100 x 100 area size with all the sensing and communication range used in this study. The blue dots represent SNs, whereas area within the circle is the communication range of the SN.

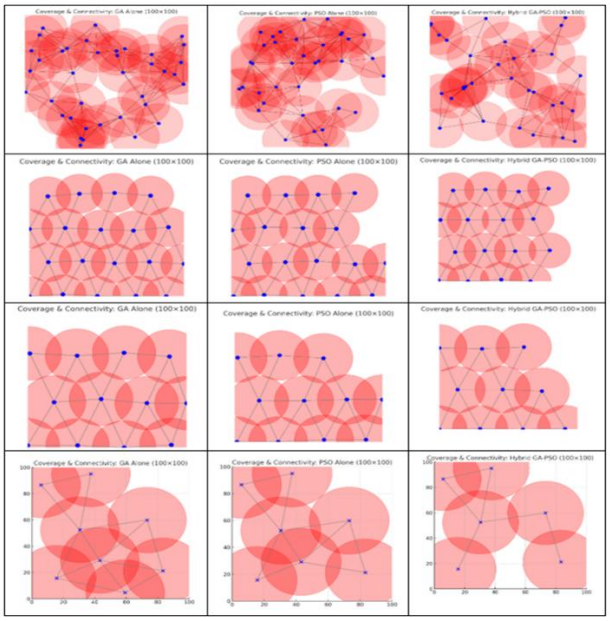


Fig. 6. Node deployment using GA, PSO and GA-PSO.

The comparison graph in Fig. 7 illustrates the differences between the set-1 and set-2 parameters for GA, PSO, and Hybrid GA-PSO. The updated results show a slight reduction in the number of deployed SNs across all methods, indicating improved efficiency. The Hybrid GA-PSO remains the most optimal approach, consistently requiring fewer nodes than GA or PSO alone. This suggests refinements in the optimization process, leading to better sensor deployment with minimized redundancy.

Further, we compared our proposed hybrid GA-PSO algorithm with another state-of-the-art multi-objective metaheuristic algorithm, CMOMPA [30-32]. The comparison is made for a large AoI with large sensing (25m) and communication range (50m). Both algorithms were implemented with the same parameters. In this comparison, GA-PSO outperforms CMOMPA due to its hybrid mechanism that strategically balances exploration (via GA's crossover/mutation) and exploitation (via PSO's velocity-driven refinement). Fig. 8 explains how GA-PSO efficiently optimizes node placement, ensuring near-optimal coverage (97 to 99%) and 100% connectivity by explicitly penalizing disconnected configurations in its fitness function. In contrast, CMOMPA's reliance on Gaussian perturbations and predator-prey dynamics often results in scattered clusters with coverage gaps (~92%) and connectivity failures (~94%), as its exploration lacks directed refinement and constraint enforcement. GA-PSO further excels in node efficiency, deploying 30 to 50% fewer sensors (e.g., 18 vs. 24 nodes for a 500m×500m area) by dynamically penalizing redundancy, while CMOMPA's unguided search increases redundancy. Statistically, GA-PSO's superiority is validated by p-values < 0.05 across coverage, connectivity, and node metrics, confirming its robustness. Fig. 9 shows the Wilcoxon test.

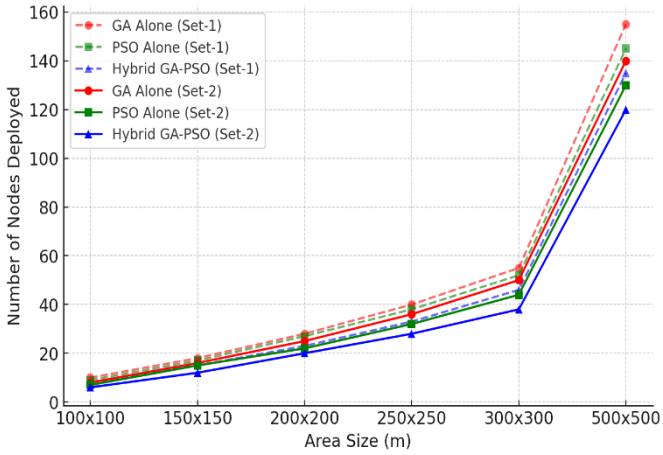


Fig. 7. Comparison of set-1 and set-2 parameters used for GA, PSO, and GA-PSO performance.

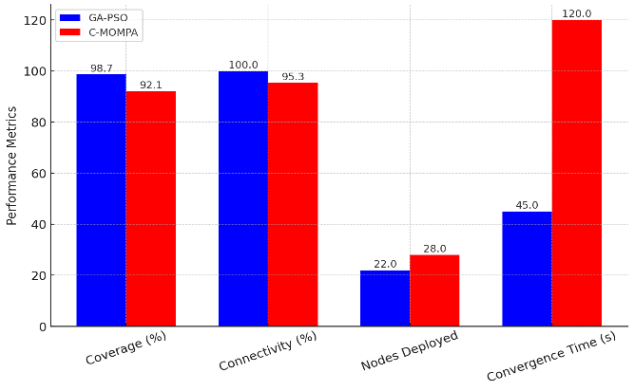


Fig. 8. GA-PSO vs CMOMPA performance comparison.

TABLE VI. SIMULATION RESULTS

Metric	Rs = 10m (GA-PSO)	Rs = 10m (C-MOMPA)	Rs = 15m (GA-PSO)	Rs=15m (C-MOMPA)	Rs=20m (GA-PSO)	Rs=20m (C-MOMPA)	Rs=25m (GA-PSO)	Rs=25m (C-MOMPA)
Coverage (%)	98	95	97	94	97	93	97	92
Connectivity (%)	100	96	100	95	100	95	100	94
Nodes Deployed	42	49	34	41	26	32	18	24
Time (s)	55	130	52	123	50	116	48	110

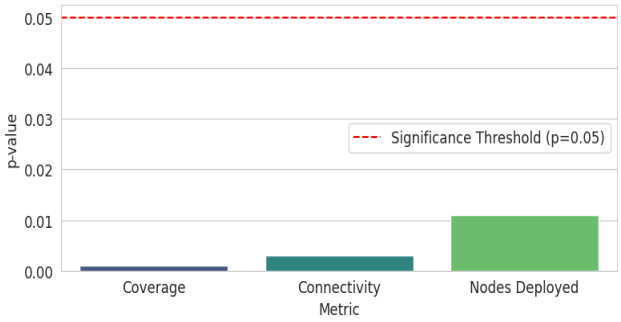


Fig. 9. Wilcoxon test result.

In the experimental setup, we used identical parameters for both algorithms, including a population size of 50, a maximum of 100 iterations, and 30 independent runs for different sensing ranges (10, 15, 20, and 25 meters). The simulation results (Table VI) demonstrate that GA-PSO consistently achieves higher coverage across all sensing ranges compared to C-MOMPA. Additionally, GA-PSO ensures 100% connectivity with fewer deployed nodes, whereas C-MOMPA struggles to maintain connectivity as the sensing range increases. Furthermore, GA-PSO exhibits faster node deployment times to enhance the understanding of the readers. Values presented in Table VI represent the mean of the generated data, with all values converted to decimal by taking the floor of the original values, whereas values in figures contain real values.

GA-PSO succeeds due to its structured search mechanism, where PSO's velocity updates systematically guide nodes toward optimal grid-like patterns, avoiding the inefficient random clustering in CMOMPA, and its constraint-aware fitness function, which enforces practicality by penalizing disconnections and excess nodes. Fig. 8 shows that GA-PSO achieves faster convergence (~48 seconds vs. CMOMPA's ~110 seconds) through PSO's social learning, making it ideal for time-sensitive deployments like IIoT, disaster management, smart cities, and environmental monitoring. However, real-world deployment presents challenges such as signal interference, obstacles, and node failures. Scalability remains a critical concern, as large-scale WSNs demand efficient parallel processing and adaptability to dynamic conditions where sensors may relocate or fail. In IIoT, the optimal placement of SNs ensures predictive maintenance, process automation, and hazard detection in industrial environments, while efficient SN deployment ensures effective data collection in disaster management. In smart cities, SNs are deployed to maximize the coverage of traffic monitoring and air quality monitoring. Applications requiring strict coverage-connectivity trade-offs and cost-sensitive scenarios prioritizing hardware minimization.

Energy constraints complicate deployment, necessitating energy-aware clustering, duty cycling, and adaptive power management to extend network lifespan. Communication issues such as packet loss, congestion, and interference require interference-aware routing and transmission control, while deploying WSNs in harsh environments calls for resilient, self-adaptive strategies, potentially leveraging reinforcement learning for dynamic reconfiguration. CMOMPA, with its unguided exploration and slower performance, remains more suitable for theoretical or loosely defined multi-objective problems. Cost-effective, large-scale deployment remains a challenge, necessitating optimization techniques that balance performance with affordability. In essence, GA-PSO dominates real-world, constrained WSN deployments by balancing efficiency and precision, while CMOMPA remains relegated to theoretical or loosely defined multi-objective problems due to its unguided exploration and slower performance.

VII. Conclusion and Future Work

The combination of GA's exploration and PSO's global search abilities allowed the hybrid algorithm to overcome the limitations of each algorithm, providing a well-balanced and effective solution for the multi-objective optimization of node deployment in WSNs. The algorithm's capability to handle the trade-offs between coverage, connectivity, and energy consumption offers a promising strategy for improving WSN design and deployment in real-world scenarios. Future research should focus on integrating deep learning with metaheuristic optimization for self-adaptive WSNs, developing lightweight algorithms for real-time decision-making, and incorporating edge computing to reduce computational overhead. Addressing cybersecurity risks and ensuring robust, low-latency communication will further enhance WSN reliability. While GA-PSO dominates real-world, constrained WSN deployments by balancing efficiency and precision, overcoming scalability, energy efficiency, and deployment challenges remains crucial for its widespread implementation in IoT and WSN applications. While the proposed hybrid GA-PSO approach has shown promising results, but requires enhancement. There are several avenues for further research and enhancement:

A. Heterogeneous WSN Deployment

In this study, we focused on homogeneous static WSNs. Future work could explore the application of hybrid metaheuristic techniques in heterogeneous WSNs, where nodes may have different sensing, communication, and energy capabilities. This would introduce additional complexity but could lead to more efficient and realistic deployment strategies.

B. Mobile WSNs

The current approach assumes static node deployment. Incorporating mobility, where sensor nodes can move to optimize coverage and connectivity over time, is an interesting direction for future research. Hybrid metaheuristics could be extended to handle dynamic node repositioning in mobile WSNs.

C. Multi-Objective Optimization

Although the hybrid GA-PSO algorithm effectively balances coverage, connectivity, and energy efficiency, future work could involve exploring more advanced multi-objective optimization techniques, such as Pareto-based approaches, to provide more flexible trade-offs between competing objectives.

D. Improving Computational Efficiency

While the hybrid approach provides superior performance, it comes at the cost of increased computational complexity. Future research could focus on optimizing the computational efficiency of the hybrid GA-PSO algorithm, perhaps by incorporating parallel processing techniques or by developing

lightweight variants for deployment in resource-constrained environments. Further, the proposed algorithm performs better than CMOMPA for a larger sensing range and enhances energy consumption. In future, this issue needs to be addressed.

E. Application to Other Network Types

The proposed method could also be applied to other types of wireless networks, such as Internet of Things (IoT) networks, where similar issues of coverage, connectivity, and energy efficiency arise. Investigating the generalization of the hybrid GA-PSO approach to these contexts could yield valuable insights.

In conclusion, the hybrid GA-PSO method presents a robust and effective strategy for optimizing node deployment in WSNs, and further exploration of its capabilities in more complex and dynamic network environments holds significant promise for future developments in this field.

REFERENCES

[1] Majid, Mamoona, et al. "Applications of wireless sensor networks and internet of things frameworks in the industry revolution 4.0: A systematic literature review." Sensors 22.6 (2022): 2087.

[2] Fahmy, Hossam Mahmoud Ahmad. "WSNs applications." Concepts, applications, experimentation, and analysis of wireless sensor networks. Cham: Springer Nature Switzerland, 2023. 67-242.

[3] Adu-Manu, Kofi Sarpong, et al. "WSN architectures for environmental monitoring applications." Journal of Sensors 2022.1 (2022): 7823481.

[4] Boardman, Nicholas T., and Kelly M. Sullivan. "Time-based node deployment policies for reliable wireless sensor networks." IEEE Transactions on Reliability 70.3 (2021): 1204-1217.

[5] Karimi-Bidhendi, Saeed, Jun Guo, and Hamid Jafarkhani. "Energy-efficient deployment in static and mobile heterogeneous multi-hop wireless sensor networks." IEEE Trans. on WC 21.7 (2021): 4973-4988.

[6] Houssein, Essam H., et al. "Metaheuristic algorithms and their applications in wireless sensor networks: review, open issues, and challenges." Cluster Computing 27.10 (2024): 13643-13673.

[7] Hua, Yicun, et al. "A survey of evolutionary algorithms for multi-objective optimization problems with irregular Pareto fronts." IEEE/CAA Journal of Automatica Sinica 8.2 (2021): 303-318.

[8] Kumar, Pawan, and S. R. N. Reddy. "Wireless sensor networks: a review of motes, wireless technologies, routing algorithms and static deployment strategies for agriculture applications." CSI Transactions on ICT 8.3 (2020): 331-345.

[9] Dongping, Dai., Jinwang, Yi. "Node Deployment Strategy Based on Improved Particle Swarm Algorithm in Three-dimensional Underwater Sensor Networks." J of P: Conf. Series (2023).

[10] Layolin, Benisto, L, C., Rajeev, Sukumaran., Agnel, Shyam, Kumar, C. "Node Deployment Strategies and Challenges in Underwater Wireless Sensor Network." IEEE ICMSCI (2024).

[11] Pavithra, R., and D. Arivudainambi. "Coverage-Aware Sensor Deployment and Scheduling in Target-Based Wireless Sensor Network." Wireless Personal Communications 130.1 (2023): 421-448.

[12] Singh, Samayveer, et al. "Learning automata based heuristics for target Q-coverage." 2020 8th Int Conf on Reliability, Infocom Technologies and Optimization (Trends & Future Directions) (ICRITO). IEEE, 2020.

[13] Paulswamy, Sathees Lingam, A. Andrew Roobert, and Kaluvan Hariharan. "A novel coverage improved deployment strategy for wireless sensor network." Wireless Personal Comm. (2022): 1-25.

[14] Priyadarshi, Rahul, and Bharat Gupta. "Area coverage optimization in three-dimensional wireless sensor network." Wireless Personal Communications 117.2 (2021): 843-865.

[15] Priyadarshi, Rahul. "Energy-efficient routing in wireless sensor networks: a meta-heuristic and artificial intelligence-based approach: comprehensive review." Archives of Computational Methods in Engineering 31.4 (2024): 2109-2137.

[16] Nematzadeh, Sajjad, et al. "Maximizing coverage and maintaining connectivity in WSN and decentralized IoT: an efficient metaheuristic-based method for environment-aware node deployment." Neural Computing and Applications 35.1 (2023): 611-641.

[17] Al-Shaikhi, Ali, and Ahmad Masoud. "Efficient, single hop time synchronization protocol for randomly connected WSNs." IEEE Wireless Communications Letters 6.2 (2017): 170-173.

[18] Mass-Sanchez, Joaquin, et al. "Weighted hyperbolic DV-hop positioning node localization algorithm in WSNs." Wireless Personal Communications 96 (2017): 5011-5033.

[19] Shirazi, Ghasem Naddafzadeh, et al. "A QoS network architecture for multi-hop, multi-sink target tracking WSNs." 2008 11th IEEE Singapore International Conference on Communication Systems. IEEE, 2008.

[20] Yi, Jiahuan, et al. "Optimal convergence nodes deployment in hierarchical wireless sensor networks: An sma-based approach." Wireless Algorithms, Systems, and Applications: 16th Int Conf, WASA 2021, Nanjing, China, June 25–27, 2021, Proc. Part III 16. Springer.

[21] Deif, D. S., and Yasser G. "Classification of wireless sensor networks deployment techniques." IEEE Comm. Sur. Tut. 16.2 (2013): 834-855.

[22] Elfouly, Fatma H., et al. "Efficient node deployment of large-scale heterogeneous wireless sensor networks." App Sci 11.22 (2021): 10924.

[23] Nikhith, K., Sripad Sriram, and C. Fancy. "Node Default Detection and Optimal Placement of Nodes in WSN." 2024 International Conference on Advances in Comp, Comm and Applied Info (ACCAI). IEEE, 2024.

[24] Kusuma, S. M., et al. "Meta Heuristic Technique with Reinforcement Learning for Node Deployment in Wireless Sensor Networks." SN Computer Science 5.5 (2024): 1-11.

[25] Sun, Peng. "Research on Optimization of Node Deployment in Wireless Sensor Networks Driven by Artificial Intelligence." Transactions on Computer Science and Intelligent Systems Research 1 (2023): 17-21.

[26] Harizan, Subash, et al. "Relay Nodes Placement Approaches in Wireless Sensor Networks: A Study." Revolutionizing Digital Healthcare Through Blockchain Technology Applications. IGI, 2023. 141-162.

[27] You, Chuiju, et al. "A Distributed Deployment Method for Wireless Sensor Networks Based on Multi Agent Systems." Journal of Electrical Systems 20.7s (2024): 1290-1297.

[28] Liu, Jinxue, and Gengxin Sun. "A deployment strategy of nodes in WSN based on “X” partition." Journal of Sensors 2022.1 (2022): 8118605.

[29] Yao, Yindi, et al. "A node deployment optimization algorithm of WSNs based on improved moth flame search." IEEE Sensors Journal 22.10 (2022): 10018-10030.

[30] Chen, Long, et al. "Balancing the trade-off between cost and reliability for wireless sensor networks: a multi-objective optimized deployment method." *Applied Intelligence* 53.8 (2023): 9148-9173.

[31] Ou, Yun, et al. "An improved grey wolf optimizer with multi-strategies coverage in wireless sensor networks." *Symmetry* 16.3 (2024): 286.

[32] Mugemanyi, Sylvère, et al. "Marine predators algorithm: A comprehensive review." *ML with Applications* 12 (2023): 100471.

[33] Ahmad, Rami, et al. "Optimization algorithms for wireless sensor networks node localization: An overview." *IEEE Access* (2024).

[34] Jaiswal, Kavita, and Veena Anand. "ESND-FA: An energy-efficient scheduled based node deployment approach using firefly algorithm for target coverage in wireless sensor networks." *International Journal of Wireless Information Networks* 31.2 (2024): 121-141.

[35] Raj, Vivek Pandiya, and M. Duraipandian. "An energy-efficient cross-layer-based opportunistic routing protocol and partially informed sparse autoencoder for data transfer in wireless sensor network." *Journal of Engineering Research* 12.1 (2024): 122-132.

[36] Binh, Huynh Thi Thanh, et al. "A heuristic node placement strategy for extending network lifetime and ensuring target coverage in mobile wireless sensor networks." *Evolutionary Int.* 17.5 (2024): 3151-3168.

[37] Alshammri, Ghalib H. "Enhancing wireless sensor network lifespan and efficiency through improved cluster head selection using improved squirrel search algorithm." *AI Review* 58.3 (2025): 79.

[38] Priyadarshi, Rahul, et al. "Hybrid Differential Evolution Algorithm for Optimal WSN Node Deployment." *World Congress in CS, Comp Eng & Applied Comp*. Cham: Springer Nature Switzerland, 2024.

Exploring Trust Management in Fog Computing: A Comprehensive Review and Future Challenges in Task Offloading

Liu Feng, Suhaidi Hassan, Mohammed Alsamman
School of Computing (SoC), University Utara Malaysia (UUM), 06010 Sintok, Kedah, Malaysia

***Abstract*—With the proliferation of data-driven services and latency-sensitive applications, fog computing has emerged as a pivotal extension of cloud infrastructure, enabling data processing and resource allocation at the network edge. However, the trustworthiness of task offloading in such decentralized and heterogeneous environments remains insufficiently explored, posing significant concerns related to system reliability, security, and performance. This review aims to address this gap by providing a comprehensive and systematic analysis of current research on trust-based task offloading in fog computing. The study investigates various trust evaluation mechanisms, categorizing them into three major paradigms: Direct Trust-based, Recommended Trust-based, and Comprehensive Trust. Through this classification, the study identifies and examines key trust-related metrics that influence offloading decisions, including task execution accuracy, trust evaluation accuracy, and evaluation latency. A critical assessment of the strengths and limitations of existing approaches reveals ongoing challenges such as dynamic trust management, scalability in large-scale networks, interoperability among diverse nodes, and resilience against malicious behaviours. Based on these insights, the study highlights pressing research opportunities and recommends the development of lightweight, adaptive, and context-aware trust frameworks capable of supporting real-time decision-making in dynamic fog environments. By synthesizing fragmented research and offering a forward-looking perspective, this review contributes a foundational reference for scholars and practitioners seeking to enhance the reliability and security of task offloading in fog computing, thereby supporting the evolution of more robust and efficient edge-based computing infrastructures.**

Keywords*—Trust management; fog computing; cloud computing; task offloading; heterogeneous networks; trust evaluation; task completion time

I. Introduction

Recently, the swift evolution of cloud computing, along with the rise of mobile smart devices and wireless networking technologies, has prompted a reassessment of computing architecture to better support mobile terminals. This adaptation aims to reduce system latency and enhance the user experience [1]. Therefore, Cisco introduced the concept of fog computing, defining it as a highly virtualized platform that shifts computational tasks from centralized cloud data centre to edge devices of the network [2]. This approach addresses the inherent limitations of the cloud computing architecture, which centralizes computing far from data sources, resulting in increased latency, congestion, reduced reliability, and augmented security vulnerabilities. Around a decade after the emergence of cloud computing, fog computing garnered extensive attention as an optimization of the existing cloud computing infrastructure [3].

Fog computing, in essence, represents a form of cloud computing with a local-oriented nature. It acts as an addition to the conventional cloud computing paradigm. In this regard, when cloud computing is paralleled to Wide Area Network (WAN) - based computing, fog computing can be compared to Local Area Network (LAN) - based computing. In a similar vein, just as the Content Delivery Network (CDN) deals with the problem of local caching within the TCP/IP framework, fog computing endeavors to solve the intrinsically local computing difficulties presented by cloud computing [4]. A pivotal moment in the development of fog computing occurred in November 2015, when industry giants such as Cisco, ARM, Dell, Intel, Microsoft, and Princeton University Edge Lab collaborated to establish the OpenFog Consortium [5]. The primary objective of this consortium was to promote and expedite the adoption of open fog computing, thereby facilitating advancements in the Internet of Things (IoT) domain. The fog computing market is anticipated to experience robust growth, with a projected Compound Annual Growth Rate (CAGR) of 67.90% during the forecast period of 2021-2026 [6]. Despite being a relatively new domain in the technological landscape, fog computing has rapidly established itself as a significant field.

Fog computing, introduced by Cisco, has emerged from the rapid evolution of cloud computing and mobile technologies. It shifts computational tasks from centralized data centres to edge devices, addressing latency, congestion, and security issues. Acting as a local extension of cloud computing, it parallels LAN to WAN. The OpenFog Consortium, formed in 2015 by industry leaders, promotes this technology and supports the Internet of Things (IoT). The fog computing market is expected to grow significantly, with a CAGR of 67.90% from 2021 to 2026, highlighting its growing importance in technology.

II. Fog Computing Concept

The architecture of fog computing is structured into a three-tier system extending from the network core to the periphery. This system comprises the cloud layer, the fog layer, and the terminal layer, as illustrated in Fig. 1. Compared to traditional computing paradigms, fog computing offers several advantages. These advantages include significantly reduced latency, conserved backbone bandwidth, enhanced support for

high mobility, extensive geographical distribution, interoperability, and reduced energy consumption [7, 8].

With the development of grid computing, cloud computing, and fog computing as a complement to cloud computing came into being. Fog computing cannot replace cloud computing and needs to work together with cloud computing. Fog computing, as an emerging computing paradigm focused on providing services close to the user side, and it doesn't require all data to be sent to the cloud computing centre. It builds an infrastructure for IoT-oriented distributed computing and provides various services, including computing, storage and network connectivity between interconnected devices at the edge of the network and cloud computing platforms in the core of the network. By extending the capabilities of cloud computing to the edge of the network, users can analyse and manage data locally.

All in all, fog computing is an innovative computing model that concentrates on providing services such as computing, communication and storage for users on network edge devices close to them. Specifically, it utilizes network devices to realize services like computing, storage and network communication between cloud servers and mobile terminal devices. By expanding the fog layer with storage and computing capabilities, the data processing and computing processes are made closer to the terminal devices, and then three main functions, namely data caching, localized computing and wireless access, are provided for users. This can not only effectively relieve the computing and storage pressure on the cloud but also significantly improve the response speed of the entire application system. Moreover, it can better fit the characteristics of high mobility of mobile terminal devices and fully meet the strict requirements of mobile applications for high traffic transmission and low-latency response.

A. Cloud, Fog and Edge Computing

Unlike cloud computing, the main goals of fog and edge computing are similar in some ways; both enable services to be closer to the user and provide lower latency [10]. Fog computing performs as a bridge between cloud computing and mobile terminal devices, building a platform with remarkable performance in computing, digital storage and network services. Therefore, cloud computing and fog computing are not independent entities; on the contrary, they interact and influence each other. In practical applications, there are several main differences among cloud computing, fog computing and edge computing. As shown in Table I, these differences span over various aspects, including latency characteristics, network characteristics, location awareness and target users [11].

B. Fog Computing Application Scenarios

Since the inception of fog computing, a wide range of scientific research institutions, internet companies, and scholars have conducted extensive studies on the subject. This collaborative effort has led to the emergence of various fog computing platforms, including OpenFog, LocalGrid, and PrismTech Vortex. These platforms facilitate the deployment of fog computing solutions across diverse sectors. Typical applications of fog computing encompass automotive networking, where vehicles communicate and share data; wireless sensor networks, which enable real-time monitoring; and intelligent building control systems that optimize energy use. Additionally, fog computing plays a crucial role in the IoT, augmented reality experiences, mobile communications, and Software Defined Networking (SDN) applications. This wide array of applications illustrates the versatility and potential of fog computing in enhancing network efficiency and performance [12].

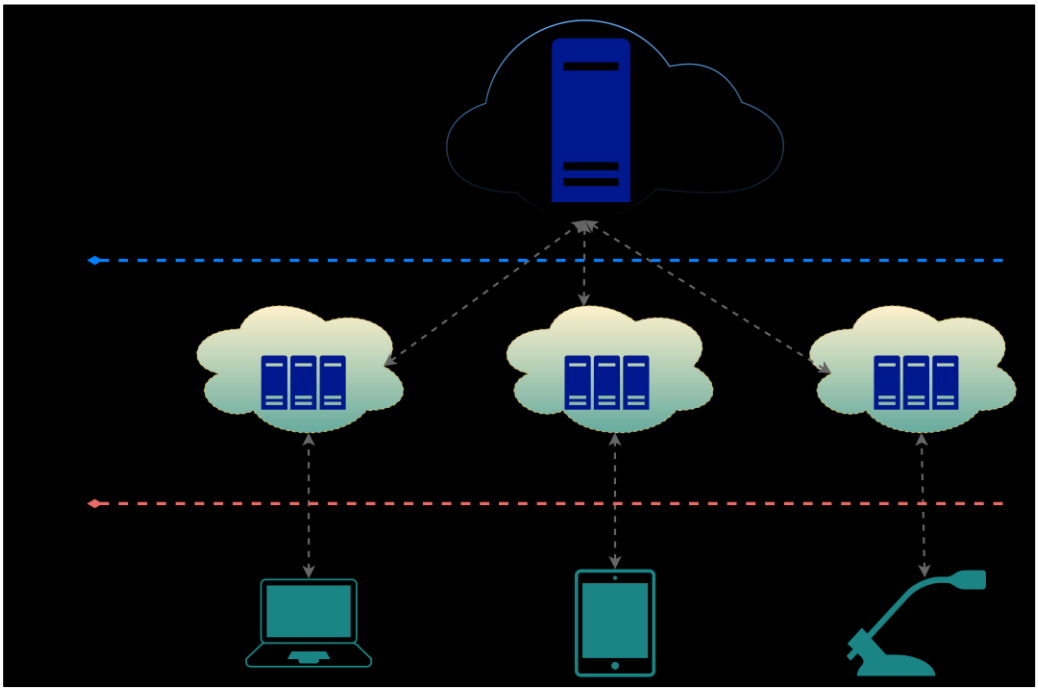


Fig. 1. Fog computing system architecture [9].

TABLE I COMPARISON ON DIFFERENT COMPUTING PARADIGMS

Attributes	Cloud Computing	Edge Computing	Fog Computing
Expected Task Execution Time	High	High-Medium Often	Low
Provided Services	Universal services	Often uses mobile networks	Vital for a particular domain and distributed
Security	Centralized (guaranteed by the Cloud provider)	Centralized (guaranteed by the Cellular operator)	Mixed (depending on the implementation)
Energy Consumption	High	Low	Varying but higher than for Edge
Identifying location	No	Yes	Yes
Main Providers	Amazon and Google	Cellular network providers	Proprietary
Interaction in Real-Time	Available	Available	Available
Latency	High	Low	Varying but higher than for Edge
Bandwidth Cost	High	Low	Low
Storage capacity and Computation	High	Very limited	Varying
Scalability	Average	High	High
Overall usage	Computation distribution for huge data (Google MapReduce), Apps virtualization, Storage of data scalability	Control of traffic, data caching, wearable applications	CCTV surveillance, imaging of subsurface in real-time, IoT, Smart city, Vehicle-to-Vehicle (V2X)

Fog computing as much as possible to the user's needs of the content and application services close to the end user, through this way, fog computing with support for mobile performance, low latency and high scalability and other advantages. Fog computing can be widely used in augmented reality (Augmented Reality, AR) and health monitoring and other real-time services; smart grid and local content distribution and other data dissemination areas; mobile big data processing and computing offloading and other distributed data areas; shopping centres and public transport and other temporary storage and other areas. In distributed data scenarios, fog computing efficiently handles user requests by leveraging the computing and storage capabilities of fog nodes. This approach alleviates the pressure on centralized cloud data processing and minimizes the response time for end-user data requests. Additionally, it reduces both the data traffic sent to the cloud and the storage requirements within it. Ultimately, this leads to a more balanced distribution of data processing demands across the cloud, fog, and terminal layers.

III. MOTIVATION

With the development of the Internet, fog computing has gradually become a new type of Internet infrastructure that has attracted much attention, but its research in the field of trusties is still insufficient [14]. Trust evaluation and management for fog nodes is an important means to improve the reliability of the network, which can effectively mitigate the security risks brought by the geographical dispersion of nodes and differences in the network environment. Although fog computing is more secure than cloud computing due to temporary data storage and local processing [15], inter-node collaboration may still trigger malicious behaviors, such as theft or manipulation of private data by malicious nodes. Traditional encryption solutions cannot effectively counter internal attacks from authenticated malicious fog nodes [16].

Trust management in fog computing faces critical challenges, particularly delayed trust value updates caused by computational overhead. While mechanisms like fuzzy logic trust evaluation, and multi-source feedback aggregation enhance trust and accuracy, they increase computational complexity, leading to high resource consumption and delayed updates in dynamic environments. Hierarchical trust models and reputation-based schemes, such as Gu et al. [17], improve reliability but introduce significant computational burdens. Similarly, adaptive mechanisms like those proposed by Almas et al. [18] struggle with timely updates due to overhead, and methods relying on recommendation information are vulnerable to malicious nodes. Fog computing faces several challenges in the process of trust management, particularly in balancing trust management and system performance. The complexity of trust models, such as reputation-based or subjective logic models, introduces significant computational overhead, making real-time decision-making difficult in large-scale environments. For instance, Alemneh et al. [19] highlight that while two-way trust management enhances security, it places a heavy computational burden on resource-constrained fog nodes. Moreover, approaches like the one proposed by Atwa et al. [20] experience difficulties in adapting to dynamic environments, such as Vehicular Ad Hoc Networks (VANETs), while models that aggregate multiple feedback sources, as discussed by Liang et al. [21], are prone to implementation complexity and require robust infrastructure. Furthermore, techniques like fuzzy logic and reputation-based mechanisms, employed by Bukhari et al. [22], introduce increased computational complexity that can hinder real-time performance. These challenges are compounded by the need for frequent updates in highly dynamic environments, which may lead to delays in trust evaluation and, consequently, suboptimal node selection.

The fog computing environment faces significant challenges related to reliability, primarily due to the participation of various vendors, including private cloud providers and multiple Internet Service Providers (ISPs). A fog network comprises numerous fog nodes, and ensuring their reliability is crucial during the task offloading. If the tasks are sent towards the malicious fog nodes, it can result in serious issues such as denial of service, jamming, spamming, impersonation, and data tampering [23, 24].

IV. CONTRIBUTION

This study advances research on trust-based task offloading in fog computing through several key contributions.

It first provides a systematic classification of trust evaluation mechanisms into three paradigms—Direct Trust-based, Recommended Trust-based, and Comprehensive Trust—consolidating fragmented research to clarify their principles and application scenarios, facilitating targeted comparisons for specific fog use cases.

Second, the work identifies and analyses critical trust-related metrics (task execution accuracy, trust evaluation accuracy, evaluation latency) and their interdependencies, offering a holistic reference for designing multi-objective trust frameworks that balance security and efficiency, critical for latency-sensitive fog applications.

Third, it critically assesses existing approaches, delineating core challenges including dynamic trust management in volatile networks, scalability in large-scale deployments, interoperability barriers, and vulnerability to malicious behaviours. This analysis transcends summarization to guide researchers toward high-impact problems.

Fourth, the review highlights emerging opportunities and recommends lightweight, adaptive, context-aware trust frameworks for real-time decision-making in dynamic fog environments, bridging research and practical implementation with actionable roadmaps.

Finally, by synthesizing knowledge into a unified perspective, it serves as a foundational reference for academia (guiding state-of-the-art advancements) and industry (supporting robust system deployment), advancing fog computing as a pivotal cloud extension.

V. ORGANIZATION

The organization of the study is designed to provide an in-depth exploration of fog computing, beginning with an Introduction that establishes the topic's significance in the context of modern technology, as shown in Fig. 2.

This section sets the stage for understanding how fog computing enhances traditional computing frameworks. Following this, the study delves into the Fog Computing Concept, where it outlines the core principles that define fog computing and its advantages over conventional cloud computing. Next, the study differentiates among Cloud, Fog, and Edge Computing, providing clarity on their unique roles and how they interact within the broader computing ecosystem. This distinction is crucial for understanding the specific contributions of fog computing. The section on Fog Computing Application Scenarios highlights various real-world implementations, showcasing its transformation influence across several industries that includes smart cities, healthcare, and industrial automation [13].

The Motivation section articulates the pressing challenges that fog computing aims to address, such as latency issues and bandwidth constraints, emphasizing its relevance in today's data-driven world. Following this, the Contribution section details the novel insights and findings that the study presents, adding to the existing body of knowledge on fog computing. An overview of the study's structure is provided in the Organization section, guiding the reader through the subsequent discussions. The main body focuses on Trust in Fog Computing, which is critical for ensuring secure operations in distributed environments. This includes an examination of Trust Management strategies, the design of a Trusted Task Offloading Architecture, and the various methods involved in the Trust Evaluation Process. The Task Offloading Process is also explained, detailing how tasks are managed within a fog framework.

The study further reviews specific techniques for Trust Evaluation in Fog Computing and presents a Comparative Analysis of Calculation Methods, evaluating different approaches to trust assessment. In addition to these discussions, the section on Classical Algorithms for Task Offloading examines traditional algorithms that facilitate efficient task management in fog environments, with a focused look at the particle swarm optimization algorithm and its relevance. A comprehensive Related Work section reviews existing literature, identifying gaps in current research and highlighting contributions to the field. The study concludes with a discussion on Future Challenges, addressing potential obstacles that may arise as fog computing evolves. Finally, the Conclusion summarizes the key findings and emphasizes the critical roles of trust and task offloading in the advancement of fog computing, reinforcing its importance in the current technological landscape.



Fig. 2. Organization of the study.

VI. Trust in Fog Computing

Trust is essential in fog computing due to its decentralized architecture and proximity to end-users, requiring robust mechanisms to ensure data integrity and reliable service delivery. This is achieved through several interconnected processes:

1) Trust management: Establishes and maintains trust relationships between nodes.

2) Trusted task offloading architecture: Ensures tasks are only delegated to trustworthy nodes.

3) Trust evaluation process: Defines methods for assessing the trustworthiness of nodes.

4) Task offloading process: Integrates trust verification prior to data transfer.

5) Dynamic trust assessment: Employs specific techniques to evaluate trust levels in real time.

6) Comparative analysis of calculation methods: Assesses the effectiveness of different trust calculation approaches.

Together, these components create a comprehensive framework that fosters a secure and reliable fog computing environment.

C. Trust Management

Fog nodes in fog computing networks are capable of executing data processing tasks near the data sources, which enhances support for real-time applications and reduces dependency on remote cloud resources. However, security and privacy concerns remain prominent within these networks.

Although fog computing is often perceived as more secure than traditional cloud computing due to the localized storage and analysis of data, vulnerabilities still exist. For instance, when fog nodes collaborate and share data to complete tasks, they may expose themselves to risks. Offloading tasks to untrustworthy fog nodes can lead to unauthorized access or manipulation of sensitive user information, raising significant privacy concerns.

To address these challenges, effective management of trust within fog nodes is vital. Implementing robust trust frameworks can help ensure that only reliable nodes are used for data processing. This includes assessing the trustworthiness of nodes, establishing secure communication channels, and continuously monitoring node behavior to prevent malicious activities. By prioritizing trustworthy management, fog computing can enhance its security posture, safeguarding user data and maintaining the integrity of services. Sunilkumar S. Manvi and others [25] explored various aspects of trust management within the realm of fog computing. Trust is a critical factor in the contemporary networked landscape, significantly influencing users' willingness to adopt fog computing solutions. In their research, they emphasize that trust management systems can build user confidence by ensuring secure and transparent data processing and storage. Effective trust management mitigates risks related to data integrity and privacy while enhancing the reliability of fog networks. By implementing clear trust protocols and assessment mechanisms, users can feel assured that their information is handled appropriately, which is crucial for the widespread adoption and success of fog computing technologies.

D. Trusted Task Offloading Architecture

As depicted in Fig. 3, this solution primarily focuses on the task offloading mechanism based on trust values within fog networks. In this architecture, every fog node has the capability to directly share tasks with other fog nodes. Meanwhile, it is assumed that fog nodes can reach each other from any location within the same fog domain. In real-world task flow scenarios, the task arrival rate can vary significantly depending on the location of the fog nodes [26].

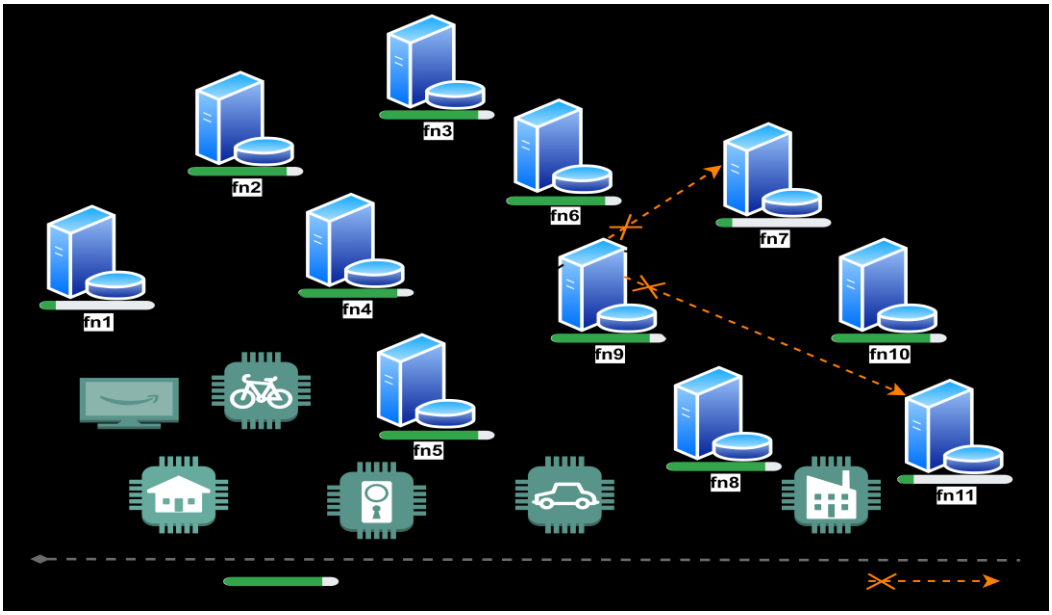


Fig. 3. Trusted task offloading architecture.

E. Trust Evaluation Process

Each fog node is equipped with a trust evaluation module that regularly assesses the trustworthiness of other fog nodes. Additionally, each node can function as both a task initiator and receiver, enabling task offloading to other fog nodes when necessary. When offloading a task, the node selects the most suitable fog node based on trust evaluation results and additional metrics such as completion time and latency [27].

Fog nodes can obtain each other's network load lightness information through local communication mechanisms. For example, network load information sharing: fog nodes can periodically exchange information about their own resource usage and task load among themselves. Such information includes CPU utilization, memory usage, task rank length, etc. The communication can be carried out through a neighbor-based communication policy. Trust evaluation is done by the fog nodes themselves. Each fog node can initiate computational detective tasks and select other fog nodes to handle these tasks based on the trust evaluation results. Fig. 4 shows the flowchart of trust evaluation, and the steps of trust evaluation can be represented as:

a) Initialization: All fog nodes set the same initial trust value.

b) Trust evaluation: Adjusting the trust value according to the completion of tasks by other fog nodes. When the network load is light, it is performed by sending some probing tasks with less data volume to the fog nodes with free resources. When the network load is large, i.e., exceeds the pre-set load threshold, the frequency of trust evaluation is reduced, or the trust value of the collaborators is evaluated by the fog nodes during the task collaboration.

c) When a fog node needs to offload a task, it identifies the most appropriate node by considering trust evaluation results and time delays. This strategy guarantees that tasks are directed to dependable fog nodes that offer optimal performance. As a result, overall efficiency and security are significantly improved.

F. Task Offloading Process

As shown in Fig. 2, there are two types of fog nodes present in the fog layer: trusted fog nodes and untrusted fog nodes. At the time of offloading, according to the different number of task packets and priority of the tasks (indicated by different colors in the figure), which are distributed to different fog nodes. Each fog node maintains a task processing rank for tasks waiting to be processed, and each task has its arrival time, processing time and deadline. The relationship between the task processing rank and the time delay is as follows: the length of the task processing rank, which represents the number of tasks in the rank, may affect the waiting time of the tasks. The longer the rank, the longer the waiting time may be, which leads to an increase in the total time delay of the task. When a fog node fn_9 initiates a task offloading interaction, it selects a neighbouring fog node to interact with. Neighboring node fn_7 has a short task rank but a low trust value and hence does not interact with node fn_7. Whereas, neighboring node fn_{10} has both low latency and high trust and hence chooses to interact with node fn_{10} on tasks.

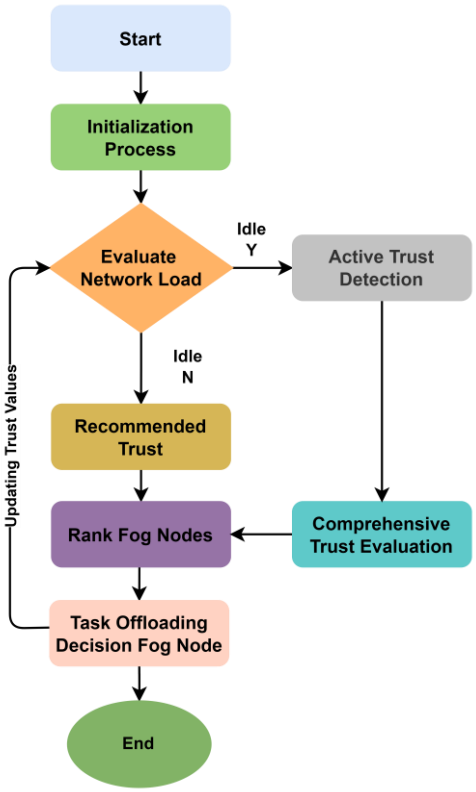


Fig. 4. Trust evaluation process [28].

G. Fog Computing Concept

Trust management in direct trust, recommended trust, and comprehensive trust:

a) Direct trust: In the fog-to-fog collaboration model, a fog node evaluates its interactions with other fog nodes, such as during task offloading or resource sharing. It updates its trust levels based on the outcomes of these interactions. For instance, the trust value of a fog node that successfully completes a task increases, while the value for a node that fails or produces subpar results decreases. The results of these trust calculations are recorded locally to assess the trustworthiness of each node.

b) Recommended trust: Fog nodes can request trustworthiness recommendations about third-party fog nodes from other fog nodes. This recommendation trust information helps fog nodes to understand the trust status of nodes in the entire network and thus make more informed task offloading decisions. However, the recommendation trust information can be affected by malicious nodes spreading false information. However, the recommended trust information may be affected by malicious nodes spreading false information. To ensure the reliability of the recommendation trust information, the trustworthiness of the recommender and the similarity of the recommendation information (recommender's trustworthiness) are calculated. Similarity (weight of recommendation trust) of the recommendation information. In this way, after obtaining direct trust and recommendation trust information about other fog nodes.

c) Comprehensive trust: The comprehensive trust value is a metric that thoroughly assesses the reliability and trustworthiness of fog nodes. It combines two key dimensions: direct trust value and recommended trust value. This integrated approach offers a solid foundation for making task offloading decisions in fog computing. By providing a clearer picture of fog nodes' performance during task allocation, the comprehensive trust value helps in selecting the most suitable nodes for task assignments.

H. Comparative Analysis of Calculation Methods

Each trust value calculation method has its unique advantages and disadvantages. Choosing a suitable calculation method and improving it with optimization suggestions can better address the challenges in specific application scenarios and improve the accuracy and efficiency of trust value calculation [20]. In Table II, combining the advantages and weaknesses of the seven algorithms mentioned above, along with their application scenarios, suggests that the fuzzy logic algorithm is particularly well-suited for real-time task offloading needs.

TABLE II MAINSTREAM WAYS OF TRUST EVALUATION

Techniques	Advantages	Weaknesses	Applicable Scenarios
Weighted Sum Method	Simple to implement and flexible	Dependent on weight settings, linear assumptions	Used in scenarios where there are fewer indicators and the weights are relatively easy to determine
Fuzzy Logic	Flexible rule definition for complex environments	Too many rules may lead to computational inefficiency	Fuzzy logic is suitable for scenarios dealing with uncertainty and ambiguity
Bayesian Inference	Dynamically updated and adaptable	Higher computational complexity	Used in scenarios where trust needs to be updated dynamically
Trust Chain Model	Highly flexible	Over-reliance on network structure	Distributed network environments for multi-hop trust delivery
Machine Learning Methods	Ability to handle complex multi-dimensional data and non-linear relationships	Requires large amounts of high-quality training data	Ideal for scenarios dealing with complex, multi-dimensional data
Graph-Based Trust Calculation	Ability to globally evaluate the trust value of nodes in the network	Slower response to dynamic changes, which may lead to lags	Suitable for social networks, P2P networks and other scenarios
Reputation-Based Trust Calculation	Flexibility to adjust the final trust value according to different evaluation sources and weights	Vulnerable to malicious ratings or subjective bias	Used in scenarios where reputation needs to be assessed and built up over time

VII. CLASSICAL ALGORITHMS FOR TASK OFFLOADING

The task of offloading problems in fog computing is typically considered as either a single-objective or multi-objective optimization problem, subject to diverse constraints. These constraints include task completion latency, the quantity of tasks fulfilled within the deadline, the degree of load balancing, the energy consumption of the fog computing system, and the overall service quality. This section outlines several commonly used algorithms for addressing task offloading challenges. Among the algorithms discussed are heuristic approaches such as the PSO algorithm and the Ant Colony (AC) algorithm, as well as the Min-Min algorithm. These methods provide crucial insights and form the basis for the research explored in this study.

A. Particle Swarm Optimization Algorithm

The PSO originated from scholars' research on the feeding behavior of bird flocks and was first proposed by Kennedy and Eberhart [29]. The particle swarm algorithm has become a hot research topic, and has been used by scholars to produce a large number of research results in various fields, and has been continuously improved to produce a number of improved PSO algorithms, such as the binary PSO algorithm, the heterogeneous PSO algorithm, the adaptive PSO algorithm, the cooperative PSO algorithm, and the discrete PSO algorithm, etc, [30]. Compared with other intelligent optimization algorithms, particle swarm optimization algorithms are easy to implement and efficient for solving combinatorial optimization problems.

The main process of the standard particle swarm optimization algorithm is as follows:

1) Initialisation: Initialise the particles in the population by assigning random initial values to their positions and velocities. The positions and velocities of the particles can be initialised according to the following equation, as shown in Eq. (1) and Eq. (2):

$$x_i = r(X_{max} - X_{min}) + X_{min} \quad (1)$$

$$v_i = r(V_{max} - V_{min}) + V_{min} \quad (2)$$

In the above equations, "r" represents a random number between 0 to 1, "X_{max}" and "X_{min}", respectively, denote the maximum and minimum positions of the particle, and "V_{max}" and "V_{min}", respectively, denote the maximum and minimum velocities of the particle. The velocity and position of the particle cannot exceed the specified maximum and minimum ranges.

The fitness values of particles within a population are determined by setting a fitness function. This fitness function should be established based on the specific objective function that requires optimization.

The fitness value of each particle's current position within the population is compared with the fitness value of its historical best position. Through this comparison, the best fitness value for each individual particle can be obtained.

The fitness values of the current positions of all particles in the population are compared with the fitness value of the global best position. As a result, the global best fitness value of the population can be determined.

Update the velocity and position of particles. For the $(t+1)$ generation iteration, update each particle's velocity and position [see Eq. (3) and Eq. (4)]:

$$v_{id}(t+1) = v_{id}(t) + c_1 r_1(t)[p_{id}(t) - x_{id}(t)] + c_2 r_2[p_{gd}(t) - x_{id}(t)] \quad (3)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (4)$$

where, r_1 and r_2 denote random numbers between (0,1), p_{id} and p_{gd} are the optimal positions of the particle and the population, respectively, c_1 and c_2 are the acceleration constants. One of the common optimization methods for particle swarm algorithms is to add inertia weights to control the current velocity of the inherited particles when updating their positions. ω includes a check to see if the maximum number of iterations has been reached, or if the desired fitness value has been reached. If the termination condition is met, the individual with the highest fitness value is output.

B. Ant Colony Algorithm

The Ant Colony Optimization (ACO) algorithm is a swarm intelligence approach inspired by the foraging behavior of ants and driven by positive feedback. When ants search for food, they release pheromones to communicate with the colony, naturally favoring shorter paths that lead to the target more quickly. As ants traverse these paths, pheromone accumulation increases, making those routes more attractive to others. Over time, pheromones evaporate, ensuring the colony gravitates towards the path with the highest concentration, ultimately leading to the most efficient solution.

Similar to the particle swarm optimization algorithm, ACO is a highly effective parallel search method capable of solving complex combinatorial optimization problems [31]. The typical process of an ACO algorithm follows these steps:

Initialize parameters: Initialize parameter information, such as colony size and pheromone.

Place ants: Randomly place ants in the colony on the path.

Path selection: The ants choose the path according to the difference of pheromone concentration on the path, and the probability of choosing the path (i,j) is Eq. (5):

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}(t)]^\beta}{\sum_{s \in J_k(i)} [\tau_{is}(t)]^\alpha \cdot [\eta_{is}(t)]^\beta}, & j \in J_k(i) \\ 0, & \text{others} \end{cases} \quad (5)$$

where, $(J_k(i) = \{1,2,\dots,n\} - tabu_k)$ is the set of next paths that the ant can choose, $tabu_k$ represents the set of paths that the ant has already travelled, and the paths that have already been chosen will be added to the taboo list for no further choices, and the ant completes a path selection when all paths have been added to the taboo list; τ_{ij} is the pheromone concentration of the path (i,j) at this point in time. Pheromone concentration; η_{ij} is the heuristic factor with $\eta_{ij} = 1/b_{ij}$, the expected degree of ant κ choosing the path (i,j). Where, α is the pheromone heuristic factor, indicating the degree of importance of the pheromone, and β indicates the degree of importance of the heuristic factor. Generally, α takes a constant value between 1 and 4, with its value positively correlated to the influence of the pheromone. To find the optimal path, once all the ants have completed their path selection, the best path among all the ants is selected. The pheromone is then updated. In this iteration, the pheromone on the path remains constant, and the update occurs according to the following Eq. (6) and Eq. (7):

$$\tau_{ij}(t+n) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij} \quad (6)$$

$$\Delta\tau_{ij} = \sum_{k=1}^{m} \Delta\tau_{ij}^k \quad (7)$$

where, $\rho(0 < \rho < 1)$denotes the evaporation coefficient of the pheromone on the path, $\Delta\tau_{ij}$ denotes the increment of pheromone in this iteration. $\Delta\tau_{ij}^k$ denotes the pheromone produced by ant k on path (i, j).

C. Min-Min Algorithm

The Min-Min algorithm was originally proposed to solve the task offloading problem in Grid computing [32]. The idea is first to map tasks with small computational resource requirements and schedule these small tasks to servers with high computational power, i.e., fast execution speed. The typical execution steps of the Min-Min algorithm are: Compute the expected completion time of every task on each available server. Then, identify the earliest completion time for each task and its corresponding server. Locate the task with the shortest earliest completion time from the task list. Dispatch this task to the server according to the earliest time calculated in Step (I).

Update the task list on the server with the earliest time. After the scheduling, update the expected completion times of other tasks in the task list on the available server, and remove this task from the task set. Repeat the above steps until all the tasks in the task list have been scheduled.

The Min-Min algorithm is simple and easy to operate, but there are some problems. On one hand, it maps many small tasks to servers with high computing power, resulting in load imbalance, and on the other hand, it causes large tasks with high computational resource demand to have no free servers to process them, so the tasks cannot be processed as quickly as possible to meet the latency requirements.

VIII. Related Works

In the increasingly complex landscape of fog computing, trust management has emerged as a critical concern, with various mechanism developed to ensure secure and reliable interactions between IoT devices and fog nodes. Mahmood et al. [33] introduced a trust-based architecture that leverages digital certificates signed by fog nodes acting as certification authorities. This approach significantly enhances the security and privacy of communications within a trusted domain, but it faces challenges related to the complexity and cost of deploying

multiple fog nodes and managing revoked certificates. Building on this, Atwa et al. [20] developed a reputation-based model for VANETs, which integrates fog computing to reduce the need for extensive cloud communication by aggregating trust evaluations at the fog nodes. Although this model reduces message transmission overhead, it encounters difficulties in adapting to the highly dynamic nature of VANET environments.

Liang et al. [21] proposed a reliable trust computing mechanism that aggregates multisource feedback within a Social Sensor Cloud (SSC) environment. By using a fusion algorithm, this approach improves the accuracy of trust evaluations and enhances the detection of malicious nodes. However, the integration of multiple feedback sources introduces significant implementation complexity and requires robust infrastructure. Meanwhile, Mahmoud et al. [34] applied the Fuzzy Analytic Hierarchy Process (Fuzzy-AHP) to prioritize multiple trust criteria, such as Quality of Service (QoS) and Quality of Security (QoSec), to deliver more accurate and reliable trust assessments. Despite its comprehensiveness, this method is limited by the complexity of implementation and the potential for bias due to reliance on expert judgment.

Alemneh et al. [19] proposed a two-way trust management system using subjective logic, which allows both service requesters and providers to evaluate each other's trustworthiness. This approach enhances the overall security of the fog network by combining direct and indirect trust, but the computational burden introduced by subjective logic, particularly in processing large volumes of recommendation information, poses challenges for resource-constrained fog nodes. In addressing cybersecurity issues within Sensor-Cloud Systems (SCS), Wang et al. [35] introduced a hierarchical trust mechanism that employs real-time monitoring and behaviour analysis at the fog layer to quickly identify and isolate malicious nodes. While effective in threat detection, this system may experience significant computational and communication overheads in large-scale networks due to the multi-layered trust computations required.

Almas et al. [18] contributed to the field by proposing a context-based adaptive trust mechanism tailored for smart healthcare systems. By utilizing a Bayesian approach and similarity measures, the system dynamically adjusts trust weights to respond to environmental changes, effectively mitigating trust-related attacks. However, the system may face challenges in maintaining timely updates to trust values in highly dynamic network environments, potentially leading to misjudgments. Bukhari et al. [22] introduced the Fog Node Selection Engine (FNSE), an AI-driven framework that employs Fuzzy Logic, Logistic Regression, and Deep Neural Networks to predict the trustworthiness of fog nodes. This approach demonstrated superior accuracy but also increased computational complexity, which could impact the system's real-time performance.

Yadav and Baranwal [36] developed a trust management mechanism based on feedback credibility evaluation to combat malicious feedback in fog computing. The system enhances the accuracy of trust evaluations by using a checkers-based method to prioritize recent interactions. However, in highly dynamic environments, the computation of trust values may not reflect the latest state of nodes, leading to delays in decision-making. Gu et al. [15] proposed a reputation-based resource allocation scheme in community-based fog computing, which matches user tasks with fog nodes that meet specific reputation criteria, thereby improving service reliability. This approach, however, introduces significant computational overhead due to the complexity of reputation computations across multiple layers of communities and nodes.

Kochovski et al. [37] integrated blockchain technology into fog computing trust management by utilizing smart contracts and trustless oracles to manage trust relationships in real-time. While this approach enhances transparency and trustworthiness, it also increases computational complexity and latency, particularly in handling a large number of transactions. Zhang et al. [38] introduced a fog-based detection system (FDS) for Sensor-Cloud Systems, focusing on hierarchical trust evaluation to detect hidden data attacks. Although effective in detecting malicious activity, the system may struggle with timely updates to trust values in dynamic networks, impacting decision accuracy.

Rehman et al. [39] proposed FogTrust, a lightweight trust management mechanism for IoT that uses fuzzy logic and trust aggregation to detect and mitigate various IoT attacks, such as on-off good-mouthing and bad-mouthing attacks. Despite its effectiveness, the multiple trust calculation processes involved, particularly with fuzzy logic, may affect the real-time performance and responsiveness of the system.

Collectively, these studies illustrate the diverse mechanisms employed to manage trust in fog computing environments. They highlight the ongoing trade-offs between enhancing security, maintaining system performance, and managing computational complexity. As the field continues to evolve, future research must address the challenges of scalability and real-time application, ensuring that trust mechanisms remain robust and efficient even in large-scale, dynamic fog computing environments.

Continuing from the previous detailed exploration, additional studies have further expanded on trust management in fog computing, focusing on various aspects such as trust evaluations, feedback credibility, and resource allocation.

Muhtadi et al. [40] presented a trust model for fog computing that leverages subjective logic to evaluate trustworthiness among fog nodes. This model considers belief, disbelief, and uncertainty in its calculations, combining direct experiences with recommendations from other nodes to assess trust. While effective in identifying and mitigating malicious nodes, the model is susceptible to bias if the network contains a high number of malicious nodes. Moreover, the reliance on recommendation information could lead to inaccuracies if the sources of these recommendations are compromised.

Mazumdar et al. [41] introduced a trust-based load-offloading protocol specifically designed to optimize service delivery in fog computing for IoT applications. This protocol uses a distributed scheme where fog nodes collaborate to share the load based on trust evaluations. Trust is established through

a Fog Registration Center (FRC) that authenticates fog nodes using Shamir's secret sharing scheme, ensuring that only trusted nodes participate in load-sharing. Although this approach minimizes service delays and ensures timely task completion, it adds complexity, especially in high-load or emergency situations where the scheme's security mechanisms might introduce additional computational burdens.

Yadav and Baranwal [42] proposed a trust evaluation method that combines Quality of Service (QoS) attributes with social relationships among fog nodes. This multi-layered approach enhances the security of fog computing environments by minimizing the risk of malicious nodes disrupting operations. However, the approach's complexity, particularly in large-scale networks, may lead to significant computational overheads and delays, as multiple layers of trust computations and data analyses are required to accurately assess trustworthiness.

Rathee et al. [43] introduced a trust-based security framework designed to enhance communication security between fog nodes and IoT devices. This framework includes a Trust Manager (TM) that evaluates the legitimacy of fog nodes and IoT devices by calculating trust values and factors based on past interactions. The TM records these metrics in a lookup table, using optimization algorithms to ensure reliable communication paths. Although the framework effectively reduces the risk of malicious activities, the need to compute multiple levels of trust values introduces high computational complexity, potentially impacting the system's real-time performance.

Ben Daoud et al. [44] developed the TACRM model, which integrates access control with trust and resource management in fog computing environments. The model computes trust based on user behaviors and assigns trust levels to users, which are then used for access control decisions. This approach enhances security by monitoring user activities and ensuring that resources are allocated securely and efficiently. However, the continuous monitoring required by the system raises potential privacy concerns, particularly in sensitive or personal data contexts.

Hamza et al. [45] proposed a bi-directional trust management system for fog computing, which uses both direct and indirect trust computations to establish trust between Service Requesters (SR) and Service Providers (SP). The system employs fuzzy logic to aggregate trust scores, with a decay function modelling the influence of past interactions on current trust assessments. While this approach effectively mitigates common trust-related attacks, such as ballot stuffing and self-promotion, the introduction of Bayesian inference and centrality into the trust evaluation process increases computational complexity, which may affect real-time performance in large-scale networks.

Ogundoyin and Kamil [16] introduced a trust management system that also uses fuzzy logic to evaluate trust in fog computing environments. This system integrates direct trust from self-observation with indirect trust from past reputation and recommendations, considering multiple criteria such as QoS and Quality of Security (QoSec). While the system combines various factors to provide a nuanced trust assessment, the use of a trust decay function in highly dynamic environments may not update trust values in a timely manner, potentially affecting the accuracy of trust-based decisions.

Finally, Rahman et al. [46] introduced a trust management system that uses a fuzzy control system to evaluate the security, reputation, and availability of vehicular fog nodes (v-fogs). This system is particularly adaptable to dynamic environments, as it handles uncertainty and variability in trust evaluation processes. The use of a fuzzy control system enables the processing of multiple inputs and the evaluation of numerous rules to generate a final trust score. However, the computational overhead introduced by this method can be challenging in real-time applications, especially in large-scale networks where rapid decision-making is crucial.

Collectively, the above literature related to trust in fog computing from 2019 to 2024 illustrates various approaches to trust management in fog computing, each addressing specific challenges related to security, reliability, and performance. In Table III, the ongoing development of these mechanisms highlights the necessity of balancing robustness with efficiency, particularly as fog computing environments scale in size and complexity. Future research in this area is likely to focus on optimizing these trust management policies to ensure that they meet the needs of real-time applications while maintaining a high level of security and scalability in heterogeneous computing environments.

TABLE III COMPARATIVE ANALYSIS OF TRUST MANAGEMENT APPROACHES IN FOG COMPUTING LITERATURE (2019-2024)

Author	Method	Advantage	Weakness	How to Evaluate
Shahid Mahmood, et al. [26]	Based on digital certificate trust management	Enhances security and privacy of data communication between IoT devices through digital certificates.	Complex and costly deployment of numerous fog nodes as certification authorities. Cumbersome management and distribution of revoked certificates across devices.	Measure latency reductions. Assess security enhancements (e.g., fewer successful cyber-attacks). Evaluate resource utilization efficiency.
Rasha Jamal Atwa,et al. [20]	Task-based Experience Reputation (TER) : combines traditional Experience-based Reputation Models and Recommendation-based Reputation Models	Significantly reduces message transmission overhead and vehicle workload compared to experience-based models.	Traditional reputation computation methods struggle to adapt to the high mobility and transient nature of vehicles in VANETs.	Performance evaluated through MATLAB simulations. Comparison against experience-based trust models. Assessment of message overhead reduction and accuracy of trust assessments via task-specific reputation values.

Junbin Liang,et al. [21]	Multi-source feedback trust computation, combining both direct and recommended trust methods	Improves detection of malicious feedback nodes, enhancing the overall trustworthiness of the SSC.	Complicated implementation due to the integration of multiple feedback sources and the need for robust infrastructure.	Performance evaluated through theoretical analysis and simulation. Simulation results demonstrate higher trust accuracy with the multisource feedback mechanism.
Ogundoyin, et al. [34]	Based on Fuzzy-AHP	Enables comprehensive evaluation of trust by simultaneously considering multiple criteria, resulting in more accurate and reliable trust assessments.	Complex implementation due to the need for expert opinions and mathematical computations for prioritizing criteria. Subjectivity in expert judgment can lead to biased results, especially with inconsistent or inaccurate opinions.	Utilizes Fuzzy-AHP to prioritize trust criteria and calculate their weights. Assesses the impact on overall trust evaluations. Effectiveness demonstrated through nuanced and accurate trust assessments, enhancing reliability and security of fog computing services.
E. Alemneh,et al. [19]	Based on Subjective Logic: Direct Trust, Indirect Trust, Trust Tuple	Ensures trustworthiness of both service providers and requesters, enhancing overall security and reliability of the fog network.	Involves complex computational operations (e.g., discounting and consensus) that may burden resource-constrained fog nodes, particularly with large amounts of recommendation information.	Performance evaluated through simulations. Focuses on metrics such as accuracy, convergence, and resistance to trust-based attacks.
T. Wang, et al. [35]	Based on Hierarchical trust: Direct Trust, Recommendation Trust, Comprehensive Trust	Quickly identifies and isolates malicious nodes through real-time monitoring.	Involves multiple levels of trust computation and data analysis, leading to high computational complexity and potential overhead in large-scale networks.	Evaluated through simulations measuring performance metrics such as energy consumption, malicious node detection speed, and recovery of misjudged nodes.
A. Almas, et al. [18]	Based on Bayesian approach: Direct Trust, Indirect Trust, Total Trust	Dynamically adjusts trust weights to respond to environmental changes, effectively mitigating trust-related attacks.	Timeliness of trust value updates may be insufficient in highly dynamic network environments, potentially leading to misjudgement.	Evaluated through simulations using Contiki-NG and Cooja in a smart healthcare scenario. Analysis includes the impact of different similarity measures and comparisons between static and adaptive weighting methods on trust scores.
S. Mahmood, et al. [33]	Based on multi-source trust evaluation: Direct Trust, Recommendation Trust, Context-Aware Feedback, Content-Based Trust	Enhances reliability and accuracy of trust assessments by incorporating multiple sources of trust data.	Involves multiple levels of trust computation and data analysis, leading to high computational complexity and potential overhead in large-scale networks.	Evaluated through extensive simulations demonstrating effectiveness and reliability in assessing user trustworthiness. Analysis includes the impact of weight factors and the significance of monitor mode.
A. A. Bukhari, et al. [22]	Based on Fuzzy Logic: Fuzzification, Rule Setup, Fuzzy Inference, Defuzzification	The FL-based FNSE approach demonstrated superior performance with the highest accuracy, precision, recall, and F1 score in predicting fog node trustworthiness.	The fuzzy logic approach involves multiple steps (fuzzification, rule inference, and defuzzification), leading to high computational complexity that may impact real-time system performance.	Conducted experiments comparing three models (FL, LR, DNN) based on metrics such as accuracy, precision, recall, F1 score, and execution time.
R. Yadav, G. Baranwal [42]	Based on Feedback Credibility Evaluation: Direct Trust, Indirect Trust, Feedback Credibility Evaluation, Final Trust Score	Filters out malicious feedback before trust evaluation, enhancing the accuracy of trust scores assigned to fog nodes.	In highly dynamic network environments, trust value computations may not reflect the latest state of nodes in a timely manner, leading to delayed decision-making.	Evaluated through simulations using synthetic data to measure effectiveness in filtering malicious feedback and maintaining accurate trust scores. Performance metrics included detection rate, false rate, and deviation of trust scores from an ideal baseline.
Ke Gu, et al. [17]	Based on Reputation Mechanism: Internal Reputation, Indirect Reputation, Overall Reputation Value	Effectively matches user tasks with fog nodes that meet reputation requirements, improving service reliability.	Involves multiple levels of reputation computation and data analysis, leading to high computational complexity, especially with many communities and nodes, which can cause overheads and delays.	Evaluated using simulations on the CloudSim platform. Focused on metrics such as reputation changes, resource allocation success rate, average completion delay of tasks, and average distance between users and service providers.
J. Al. Muhtadi, et al. [40]	Based on Subjective Logic: Direct trust, Recommended Trust, trust convergence	Effectively identifies and mitigates malicious nodes using subjective logic that incorporates uncertainty and indirect evidence.	Heavily relies on recommendation information, which can bias trust calculations in networks with a high number of malicious nodes.	Evaluated through simulations assessing accuracy, convergence, and resilience against various trust-based attacks.

N. Mazumdar, et al. [41]	Based on Direct Trust, Indirect Trust, Comprehensive trust evaluation	Minimizes service delays and ensures timely task completion by offloading tasks to trusted nodes.	Shamir's secret sharing scheme adds complexity and burden, particularly in high-load or emergency situations.	Evaluated through simulations focusing on average latency, service response rate, and task completion success.
R. Yadav and G. Baranwal, [36]	Based on multilevel trust: Trust Factor Calculation, Overall, Trust Evaluation	Enhances security in fog computing by evaluating trust through QoS and social relationships, minimizing risks from malicious nodes.	Involves multi-layered trust computation and data analysis, leading to high computational complexity and potential overheads in large-scale networks.	Considers historical interactions and reputation among fog nodes to assess trustworthiness.
P. Kochovski, et al. [37]	Based on blockchain: Smart Contracts, Trustless Smart Oracles, Markov	Supports real-time monitoring and trust updates for quick responses to changes in node behavior.	Use of blockchain and smart contracts increases computational complexity and latency, especially with numerous transactions.	Evaluated through simulations focusing on the ability to detect and handle malicious nodes while maintaining performance. Metrics included accuracy in detecting trust anomalies and system overhead.
G. Rathee, et al. [43]	Based on Tidal Trust Algorithm	Ensures only trusted nodes communicate, effectively reducing the risk of malicious activities.	Requires computation of multiple trust values and uses complex algorithms, leading to high computational complexity that may impact real-time performance.	Examines performance in scenarios with malicious versus trusted fog nodes, focusing on managing security for handoff IoT devices (HEU) and mobile HEU (MHEU).
W. Ben Daoud, et al. [44]	Based on TACRM framework	Monitors and controls access based on trust levels, reducing the risk of unauthorized access and attacks.	Continuous monitoring of user behaviour may raise privacy concerns.	Evaluated through simulations of various network scenarios to measure the impact on security breach incidents.
M. Hamza, et al. [45]	based on a Bayesian inference model: Direct Trust, Reputation Function, Degree Centrality, Service Score, Final Trust	Detects and mitigates common trust-related attacks like ballot stuffing and self-promotion.	Comprehensive trust evaluation and the use of Bayesian inference increase computational complexity, potentially affecting real-time performance in large-scale networks.	Measures the accuracy of trust assessments through modelling.
S. O. Ogundoyin and I. A. Kamil [16]	Based on fuzzy logic: Direct Trust, Indirect Trust, trust convergence	Combines QoS, QoSec, and social interactions for a nuanced trust assessment.	Trust decay function may not update values timely in highly dynamic environments, affecting decision accuracy.	Evaluated through simulations measuring the system's ability to mitigate trust-related attacks and the accuracy of trust assessments.
G. Zhang, et al. [38]	Based on Hierarchical Trust Evaluation: Direct Trust, Indirect Trust, Hierarchical Trust Evaluation	Efficiently detects hidden data attacks by analysing sensor data correlations and historical trust states.	Layered architecture may lead to untimely updates of trust values in dynamic networks, affecting decision-making accuracy.	Evaluated through MATLAB simulations testing detection accuracy of the FDS under various scenarios, including different malicious sensor ratios and group sizes.
A. Rehman, et al. [39]	Based on Fuzzy Logic: Direct Trust, Indirect Trust, Trust Encryption and Aggregation	Effectively detects and mitigates various IoT attacks, including on-off, good-mouthing, and bad-mouthing attacks through encryption and trust aggregation.	Involves complex trust calculations, particularly with fuzzy logic, leading to high computational complexity that may affect real-time responsiveness.	Evaluated through simulations testing FogTrust's effectiveness against various attacks.
Q. Zhang, et al. [47]	a semantic-based trust management system integrated: Semantic Analysis, Direct Trust, Indirect Trust	Enables faster and more accurate emergency responses through semantic analysis, reducing failure risks in critical situations.	Scaling the system may challenge trust evaluations and semantic data processing, impacting performance.	Fog nodes assess trustworthiness based on historical interactions, data freshness, and recommendations, filtering out untrustworthy content.
T. Wang, et al. [35]	based on both direct trust and indirect trust	Utilizes both direct and indirect trust to adapt to network changes, including node mobility and varying trust levels.	Growing network size may complicate trust value calculations and data management, impacting scalability.	Final trust value is computed by combining direct and indirect trust values using a weighted sum.
F. H. Rahman, et al. [46]	Based on a fuzzy control system: Security, Reputation, Availability	Fuzzy control system effectively handles uncertainty and variability in trust evaluation, adapting to dynamic environments.	Introduces computational overhead from processing multiple inputs and evaluating numerous rules, challenging real-time applications.	Trust value is calculated using a fuzzy control system that processes security, reputation, and availability metrics to determine vehicle cluster membership.

IX. Challenges and Future Prospects

As the landscape of fog computing continues to shift, new challenges emerge that must be addressed to ensure effective trust management and task offloading. The increasing complexity and dynamism of network environments require innovative solutions to enhance the reliability and security of these systems. This section outlines several key challenges and

research directions that researchers and practitioners will need to tackle in the pursuit of more robust trust management frameworks.

A. AI and Machine Learning Integration

Integrating AI and machine learning techniques can significantly enhance trust assessments by leveraging historical data and real-time interactions. Research can explore supervised and unsupervised learning algorithms to model node behavior and predict trustworthiness, allowing for more accurate and adaptive trust evaluations that evolve with the network.

B. Hybrid Trust Models

Hybrid models that combine various trust evaluation techniques can provide more resilient systems. Research can explore the integration of reputation systems, subjective logic, and Bayesian approaches to create comprehensive models that account for both direct experiences and indirect recommendations, enhancing the accuracy of trust assessments.

C. Context-Aware Trust Management

Context-aware mechanisms can adjust trust evaluations based on environmental factors and user behaviors. Future research can focus on developing systems that dynamically modify trust metrics according to contextual information, ensuring that trust assessments are relevant and responsive to the current state of the network.

D. Decentralized Trust Management Systems

Decentralized approaches reduce reliance on central authorities and enhance resilience. Research should explore the design of decentralized trust management frameworks that empower nodes to independently evaluate trust, using peer-to-peer communication protocols to share and validate trust information among nodes.

E. Cross Domain Trust Management

Managing trust across different domains (e.g., fog and cloud computing) is vital for seamless interactions. Future studies should investigate methods for establishing trust relationships that span multiple environments, ensuring that resources can be shared securely and efficiently without compromising the integrity of trust evaluations.

F. Performance Optimization Techniques

Optimizing computational processes involved in trust evaluations can minimize overhead. Research should focus on developing algorithms that streamline trust computations, utilizing techniques such as caching trust values, batching evaluations, or employing approximation methods to reduce computational complexity without sacrificing accuracy.

G. User Centric Trust Models

Developing user-centric trust management systems can enhance user experiences by prioritizing individual preferences and interactions. Research should explore personalized trust models that adapt to user behaviors and preferences, allowing for more relevant interactions and trust assessments that reflect user needs.

By addressing these challenges and pursuing the outlined research directions, the field of trust management in fog computing can evolve to meet the demands of increasingly complex and dynamic environments, ensuring both security and performance.

X. Conclusion

This study provides a comprehensive exploration of the critical role of trust management in fog computing, particularly in ensuring secure and efficient task offloading among diverse IoT devices. It highlights significant gaps in existing methodologies and emphasizes the urgent need for advanced trust assessment frameworks. By establishing key metrics for evaluating trust management systems, we lay the groundwork for measuring their effectiveness, which is essential for fostering reliable interactions in dynamic environments. The integration of AI and machine learning techniques is examined as a promising approach to enhancing trust evaluations. These technologies allow systems to adapt to real-time data and evolving network conditions. However, several challenges persist, including scalability, the management of diverse IoT devices, the achievement of real-time trust evaluations, and the need to address security and privacy concerns. Moreover, addressing these challenges through innovative research and practical implementations is crucial for advancing fog computing technologies. This work aims to provide valuable insights that will facilitate the development of more secure and efficient applications, ensuring that trust management evolves alongside the rapidly changing landscape of IoT and fog computing.

References

[1] Y. Meng, M. A. Naeem, A. O. Almagrabi, R. Ali, and H. S. Kim, "Advancing the state of the fog computing to enable 5g network technologies," Sensors, vol. 20, no. 6, p. 1754, 2020.

[2] M. A. Al-Shareeda, A. A. Alsadhan, H. H. Qasim, and S. Manickam, "The fog computing for internet of things: review, characteristics and challenges, and open issues," Bulletin of Electrical Engineering and Informatics, vol. 13, no. 2, pp. 1080-1089, 2024.

[3] R. Das and M. M. Inuwa, "A review on fog computing: Issues, characteristics, challenges, and potential applications," Telematics and Informatics Reports, vol. 10, p. 100049, 2023.

[4] M. A. Al-Tarawneh, "Bi-objective optimization of application placement in fog computing environments," Journal of Ambient Intelligence and Humanized Computing, vol. 13, no. 1, pp. 445-468, 2022.

[5] C. Arivazhagan and V. Natarajan, "A Survey on Fog computing paradigms, Challenges and Opportunities in IoT," in 2020 international conference on communication and signal processing (ICCSP), 2020, pp. 0385-0389: IEEE.

[6] G. K. Walia, M. Kumar, and S. S. Gill, "AI-empowered fog/edge resource management for IoT applications: A comprehensive review, research challenges, and future perspectives," IEEE Communications Surveys & Tutorials, vol. 26, no. 1, pp. 619-669, 2023.

[7] M. Muneeb, K.-M. Ko, and Y.-H. Park, "A fog computing architecture with multi-layer for computing-intensive iot applications," Applied Sciences, vol. 11, no. 24, p. 11585, 2021.

[8] Hassan, S., & Alsamman, M. (2021, January). Energy-aware multipath forwarding mechanism for named data network in wireless environment. In Proc. 6th Int. Conf. Internet Appl. Protoc. Serv. Endur. Internet's Use Growth Pandemic Era (pp. 43-49).

[9] K. M. Sadique, R. Rahmani, and P. Johannesson, "Fog computing for trust in the Internet of Things (IoT): A systematic literature review," in 2020 International Conference on Computer Science, Engineering and Applications (ICCSEA), 2020, pp. 1-6: IEEE.

[10] S. U. Khan, "The curious case of distributed systems and continuous computing," IT Professional, vol. 18, no. 2, pp. 4-7, 2016.

[11] A. Ometov, O. L. Molua, M. Komarov, and J. Nurmi, "A survey of security in cloud, edge, and fog computing," Sensors, vol. 22, no. 3, p. 927, 2022.

[12] P. Hu, S. Dhelim, H. Ning, and T. Qiu, "Survey on fog computing: architecture, key technologies, applications and open issues," Journal of network and computer applications, vol. 98, pp. 27-42, 2017.

[13] M. Alsamman, Y. Fazea, F. Mohammed and M. A. M. Kehail, "Fog Computing in Smart Cities: A Systematic Review," 2023 3rd International Conference on Emerging Smart Technologies and Applications (eSmarTA), Taiz, Yemen, 2023, pp. 1-8, doi: 10.1109/eSmarTA59349.2023.10293505.

[14] Z. Ashi, M. Al-Fawa'reh, and M. Al-Fayoumi, "Fog computing: security challenges and countermeasures," Int. J. Comput. Appl, vol. 175, no. 15, pp. 30-36, 2020.

[15] S. Parikh, D. Dave, R. Patel, and N. Doshi, "Security and privacy issues in cloud, fog and edge computing," Procedia Computer Science, vol. 160, pp. 734-739, 2019.

[16] S. O. Ogundoyin and I. A. Kamil, "A trust management system for fog computing services," Internet of Things, vol. 14, p. 100382, 2021.

[17] K. Gu, L. Tang, J. Jiang, and W. Jia, "Resource allocation scheme for community-based fog computing based on reputation mechanism," IEEE Transactions on Computational Social Systems, vol. 7, no. 5, pp. 1246-1263, 2020.

[18] A. Almas, W. Iqbal, A. Altaf, K. Saleem, S. Mussiraliyeva, and M. W. Iqbal, "Context-based adaptive Fog computing trust solution for time-critical smart healthcare systems," IEEE Internet of Things Journal, vol. 10, no. 12, pp. 10575-10586, 2023.

[19] E. Alemneh, S.-M. Senouci, P. Brunet, and T. Tegegne, "A two-way trust management system for fog computing," Future Generation Computer Systems, vol. 106, pp. 206-220, 2020.

[20] R. J. Atwa, P. Flocchini, and A. Nayak, "A fog-based reputation evaluation model for VANETs," in 2021 International Symposium on Networks, Computers and Communications (ISNCC), 2021, pp. 1-7: IEEE.

[21] J. Liang, M. Zhang, and V. C. Leung, "A reliable trust computing mechanism based on multisource feedback and fog computing in social sensor cloud," IEEE Internet of Things Journal, vol. 7, no. 6, pp. 5481-5490, 2020.

[22] A. A. Bukhari and F. K. Hussain, "Fuzzy logic trust-based fog node selection," Internet of Things, vol. 27, p. 101293, 2024.

[23] V. Meena, M. Gorripatti, and T. Suriya Praba, "Trust enforced computational offloading for health care applications in fog computing," Wireless Personal Communications, vol. 119, no. 2, pp. 1369-1386, 2021.

[24] B. Premalatha and P. Prakasam, "TwI-FTM: Two-way IoT-FoG trust management scheme for task offloading in IoT-FoG networks," Results in Engineering, vol. 22, p. 102197, 2024.

[25] S. S. Manvi and N. C. Gowda, "Trust management in fog computing: a survey," in Applying integration techniques and methods in distributed systems and technologies: IGI global, 2019, pp. 34-48.

[26] A. Shakarami, A. Shahidinejad, and M. Ghobaei-Arani, "A review on the computation offloading approaches in mobile edge computing: A game-theoretic perspective," Software: Practice and Experience, vol. 50, no. 9, pp. 1719-1759, 2020.

[27] M. Yu, A. Liu, N. N. Xiong, and T. Wang, "An intelligent game-based offloading scheme for maximizing benefits of IoT-edge-cloud ecosystems," IEEE Internet of Things Journal, vol. 9, no. 8, pp. 5600-5616, 2020.

[28] W. Mo, T. Wang, S. Zhang, and J. Zhang, "An active and verifiable trust evaluation approach for edge computing," J. Cloud Comput., vol. 9, no. 1, 2020, doi: 10.1186/s13677-020-00202-w.

[29] J. Kennedy and R. Eberhart, "Particle swarm optimization," in Proceedings of ICNN'95-international conference on neural networks, 1995, vol. 4, pp. 1942-1948: ieee.

[30] T. M. Shami, A. A. El-Saleh, M. Alswaitti, Q. Al-Tashi, M. A. Summakieh, and S. Mirjalili, "Particle swarm optimization: A comprehensive survey," Ieee Access, vol. 10, pp. 10031-10061, 2022.

[31] Prado-Rodríguez, R., González, P., Banga, J. R., & Doallo, R. (2024). Improved cooperative Ant Colony Optimization for the solution of binary combinatorial optimization applications. Expert Systems, 41(8), e13554.

[32] Liu, F., & Guo, W. (2019). Optimized Min-Min Dynamic Task Scheduling Algorithm in Grid Computing. Advances in Intelligent Systems and Computing. https://doi.org/10.1007/978-3-030-25128-4_92.

[33] S. Mahmood, A. Ullah, and A. K. Kayani, "Fog computing trust based architecture for internet of things devices," International Journal of Computing and Communication Networks, vol. 1, no. 1, pp. 18-25, 2019.

[34] S. O. Ogundoyin and I. A. Kamil, "A Fuzzy-AHP based prioritization of trust criteria in fog computing services," Applied Soft Computing, vol. 97, p. 106789, 2020.

[35] T. Wang, G. Zhang, M. Z. A. Bhuiyan, A. Liu, W. Jia, and M. Xie, "A novel trust mechanism based on fog computing in sensor–cloud system," Future Generation Computer Systems, vol. 109, pp. 573-582, 2020.

[36] R. Yadav and G. Baranwal, "An efficient trust management using feedback credibility evaluation method in fog computing," Simulation Modelling Practice and Theory, vol. 120, p. 102610, 2022.

[37] P. Kochovski, S. Gec, V. Stankovski, M. Bajec, and P. D. Drobintsev, "Trust management in a blockchain based fog computing platform with trustless smart oracles," Future Generation Computer Systems, vol. 101, pp. 747-759, 2019.

[38] T. Li, G. Huang, S. Zhang, and Z. Zeng, "NTSC: a novel trust-based service computing scheme in social internet of things," Peer-to-Peer Networking and Applications, vol. 14, pp. 3431-3451, 2021.

[39] A. Rehman, K. A. Awan, I. Ud Din, A. Almogren, and M. Alabdulkareem, "FogTrust: fog-integrated multi-leveled trust management mechanism for internet of things," Technologies, vol. 11, no. 1, p. 27, 2023.

[40] J. Al Muhtadi, R. A. Alamri, F. A. Khan, and K. Saleem, "Subjective logic-based trust model for fog computing," Computer Communications, vol. 178, pp. 221-233, 2021.

[41] N. Mazumdar, A. Nag, and J. P. Singh, "Trust-based load-offloading protocol to reduce service delays in fog-computing-empowered IoT," Computers & Electrical Engineering, vol. 93, p. 107223, 2021.

[42] R. Yadav and G. Baranwal, "Trust-aware framework for application placement in fog computing," in 2019 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS), 2019, pp. 1-6: IEEE.

[43] G. Rathee, R. Sandhu, H. Saini, M. Sivaram, and V. Dhasarathan, "A trust computed framework for IoT devices and fog computing environment," Wireless Networks, vol. 26, pp. 2339-2351, 2020.

[44] W. B. Daoud, M. S. Obaidat, A. Meddeb-Makhlouf, F. Zarai, and K.-F. Hsiao, "TACRM: trust access control and resource management mechanism in fog computing," Human-centric Computing and Information Sciences, vol. 9, pp. 1-18, 2019.

[45] M. Hamza, W. Iqbal, A. Ahmad, M. Babar, and S. Khan, "A social qualitative trust framework for Fog computing," Computers and Electrical Engineering, vol. 102, p. 108195, 2022.

[46] F. H. Rahman, S. S. Newaz, T. W. Au, W. S. Suhaili, and G. M. Lee, "Off-street vehicular fog for catering applications in 5G/B5G: A trust-based task mapping solution and open research issues," IEEE Access, vol. 8, pp. 117218-117235, 2020.

[47] Q. Zhang, J. Wu, M. Zanella, W. Yang, A. K. Bashir, and W. Fornaciari, "Sema-IIoVT: Emergent semantic-based trustworthy information-centric fog system and testbed for intelligent internet of vehicles," IEEE Consumer Electronics Magazine, vol. 12, no. 1, pp. 70-79, 2021.

AraSpam: A Multitask Deep Neural Network for Spam Detection in Arabic Twitter

Lulua Alhamdan[1], Ahmed Alsanad[2], Nora Al-Twairesh[3]
Department of Information Systems, College of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University, Riyadh, Saudi Arabia[1]
College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia[1, 2, 3]

***Abstract*—Twitter has become widely used for disseminating information across the Arab world. It provides diverse communicative and informational needs while serving as a rich data source for a wide range of research. However, the integrity of such data is frequently undermined by the pervasive issue of spam. Existing research proposed the use of spam detection models at multiple levels—the account, tweet, and campaign levels. Many of these models target Uniform Resource Locator (URL)-based spam messages, whereas a significant portion of spam content operates without embedded URLs. Furthermore, spam detection methodologies tailored to the account level often lack the precision required for tweet-level analysis or, conversely, fail to capture broader account-level behavioral patterns. Moreover, studies focusing on Arabic spam have largely been restricted to specific geographical regions or linguistic varieties, such as Arabic dialect (AD) or Modern Standard Arabic (MSA), thereby neglecting the full spectrum of Arabic's linguistic diversity in spam messages. This study aims to address these limitations by proposing AraSpam, a multitask deep neural network that detects both spam messages and profiles using a single model. It was trained using a dataset of tweets written in AD and MSA covering different spamming targets. The text features were extracted using transformer-based models: AraBERT for tweet text and mBERT for profile screen name. The experiment demonstrated 96% accuracy in detecting both spam accounts and tweets with seven different spamming targets. Additionally, the experiments revealed that reducing the number of spam classes resulted in an increase in tweet detection performance and a decrease at the account level.**

***Keywords*—Spam detection; Twitter; multitask deep neural network; transformer-based model**

I. Introduction

Spam is generally referred to as unwanted and unsolicited material. It takes on several forms on different platforms, including email, Wikipedia, e-commerce, and online social networks (OSNs). Recently, artificial intelligence (AI) has enabled multimedia spam and social spambots to generate automatic content [1]. Users who send spam messages are known as spammers. They could be individuals, campaigns, or social bots. They are typically motivated by a clear objective, concentrating on its attainment [2][3].

Twitter is recognized as an important OSN platform for daily activities and is employed by individuals, organizations, and researchers for information dissemination and analysis. It allows users to share messages known as tweets. A tweet is generally limited to 280 characters, including text, Uniform Resource Locators (URLs), multimedia, emojis, hashtags, and mentions. Recently, users who subscribe to premium accounts can share messages that are extended to a maximum of 25,000 characters. Twitter defines spam as "*sharing or posting content in a bulk, duplicative, irrelevant, or unsolicited manner that disrupts people's experience* [4]". The powerful information-sharing capabilities of Twitter, including hashtags, mentions, and retweets, attract spammers who exploit the platform for their activities. Commonly, spam is produced on Twitter through mentions of well-known users' accounts, replies to their tweets, or wildly trending hashtags [5][6].

Different types of spammers with various intents have been observed. On the political side, some campaigns disseminate agendas to influence people's opinions during critical periods of political significance [7][8]. Commercially, in addition to standard advertisement spam, spammers try to influence people's opinions toward specific products or services by writing fake reviews [9][10]. In cybersecurity, Twitter has become a target for attackers seeking to spread malware [11][12]. Additionally, illegal and pornographic materials are also propagated on Twitter [13][14]. Accordingly, these spamming activities are decreasing the quality of Twitter data. The decrease in data quality affects researchers by requiring additional cleaning effort before performing various tasks, including data analysis, training natural language processing (NLP) models, and others [9][15]. Twitter has implemented restrictions to ensure the reliability of its content and to prevent any deceptive practices, thereby increasing its audience's confidence in its authority [4]. However, these regulations and rules lack detection capabilities except for limited violations. Researchers have proposed different detection approaches for spam tweets, profiles, campaigns, and bots. Most rely on supervised machine learning (ML) techniques using single-type or hybrid features, including text, content, account, and graph features [16][17]. The deep learning (DL) approach eliminates the need for feature selection; however, a large labeled dataset is required for improved performance [18]. Even if high accuracy is achieved through these approaches, spammers continue to change their behaviors and challenge the detection process, a problem referred to as spam drift [19]-[21]. Regarding detection level, tweet-level detection lacks the ability to detect spamming behaviors, whereas account-level detection models cannot identify spam tweets generated from legitimate accounts. On the other hand, campaign- and bot-level detection models are unable to detect individual and ordinary spammers' accounts, respectively [16].

In Arabic language, specifically, various studies have revealed that the effectiveness of the features in the detection process is language-dependent and differs between societies [22]-[25]. At the same time, most of the literature is based on the English language [25]-[27]. Among the limited Arabic studies, the majority focused on content in Saudi Arabia [10].

This study aims to enrich Arabic Twitter spam detection by addressing different limitations. First, using a more comprehensive dataset that covers Arabic spam messages, even written in Modern Standard Arabic (MSA) or Arabic dialect (AD). Additionally, in order to eliminate the spam drift problem, the data was collected using different time slots, hashtags, user mentions, and keywords. Furthermore, unlike previous works that automatically assigned the same label to both accounts and tweets, tweets and accounts were labeled independently to highlight instances where legitimate accounts and vice versa occasionally wrote spam messages. Moreover, to reveal the purpose of spamming, spam tweets were categorized based on the most common spamming purposes on Arabic Twitter. Finally, a detection model is proposed to classify both tweets and user profiles within a single model by leveraging the concept of a multitask learning framework. Accordingly, this will help in revealing spam messages generated from legal accounts and vice versa. Furthermore, by leveraging the similarities between the two tasks, we can reduce the processing time.

The rest of this study is structured as follows: Section II presents a review of related work on Twitter spam detection. Section III provides background on the techniques employed in this study and outlines the proposed methodology. Section IV presents and discusses the experimental results. Section V presents the conclusion, summarizing the key findings and highlighting the overall significance of the study.

II. Related Work

Various methods have been proposed for spam detection on Twitter, each implemented through different techniques. Recently, the primary approaches in the literature are ML and DL, applied at different levels, including tweets, accounts, campaigns, and social bots. ML-based studies vary in their choice of features, with some focusing on a single feature type, such as text, content, user, or graph, while others adopt hybrid features. The next section explores studies employing tweet- and account-level detection approaches, with a dedicated section on Arabic spam detection research.

A. Tweet-Level Detection

First, Tweet-level detection involves classifying tweet messages into spam or non-spam. Studies on the ML approach most often use content features, as classified by [19], into URL or hashtag features, image or video features, and text-based features. Other studies considered user account features in tweet classification, such as in [28], which proposed a framework based on ML with a combination of user and tweet features that were available for real-time detection, such as the number of characters, digits, hashtags, and followers.

Recent studies have proposed extracting text features using various methods, including term frequency–inverse document frequency (TF–IDF), bag-of-words, n-gram, and Word2Vec. The authors of [18] proposed using Word2Vec to learn message syntax with a multilayer perceptron artificial neural network (MLP) that outperforms other feature-based ML classifiers such as random forest (RF) and naïve bayes (NB). The authors of [29] presented another means of outperforming these methods, using bidirectional long short-term memory networks (BiLSTMs) to extract text features for training supervised ML classifiers.

Instead of depending on feature-based ML classifiers, a variety of studies proposed models based on deep neural networks. Neha proposed using long short-term memory (LSTM) after extracting text features by applying Global Vector (GloVe) word embeddings [30], achieving an accuracy of approximately 95%. Moreover, [31] used the Word2Vec embedding for training an extreme learning machine (ELM) with multiple layers, achieving an accuracy of 88%. The use of transformer-based models for generating embeddings was applied by [32] for detecting spam in a short message service (SMS) dataset. This is accomplished by extracting text features using Generative Pre-trained Transformer 3 (GPT-3) and ensemble classifiers with weighted voting for classification.

B. Account-Level Detection

Account-level detection involves classifying a user account as a spammer or a legitimate user. Most studies accomplished it using ML with a combination of account, content, and graph features. The majority used statistical features, such as the ratio between the number of followers and the number of friends, the average number of favorite tweets, tweeting frequency, and account age [33]. In addition to these features, other studies have focused on interaction features in relation to user and content features [34]. Another study [35] analyzed URLs and sharing patterns to detect spam accounts. Detecting spammers' behaviors and extracting useful behavioral features have led researchers to perform network analyses and utilize graph features, such as average neighborhood followers, average neighbor tweets, and betweenness centrality of mentions [36]. Instead of relying on such features, [37] utilized a CNN to combine text features extracted using Word2Vec with common account numerical features, such as account age, number of followers and followings, and others. In various studies, suspicious words were combined with other features to characterize spammer accounts. Some approaches involve checking for the presence of words from a predefined dictionary [23], whereas others extract statistical features based on these words [36]. The authors of [38] utilized additional linguistic features by targeting spammers based on their topics of interest, arguing that information from individuals discussing various topics is less trustworthy than information from those focused on specific areas.

All previous approaches required a labeled dataset for Twitter accounts; however, complete profile datasets were not publicly available due to privacy concerns [39]. Several studies have proposed using the unsupervised approach to mitigate the cost of manual annotation [40]-[42]. Instead of detecting individual human spam accounts, various studies focused on detecting automated spam via social bots [49][50], while others were more interested in identifying groups of spammers forming a campaign [43]-[46].

C. Spam Detection on Arabic Twitter

Several studies have focused on determining the effect of the language used on spam classification. For example, [22] used an ML classifier on two datasets: English and Roman Urdu. They observed that the results improved when a language-dependent model was used. Another study, by the authors of [25], examined the differences between spammers in various social contexts that affect the effectiveness of selected features. They built a model for detecting Arabic spammers and tested it with two datasets—Arabic accounts and English accounts—and concluded that the accuracy decreased in detecting English spammers. The authors of [23] tested the effectiveness of features in English, Arabic, Korean, and Spanish datasets and found that the behavior of spammers and the effectiveness of features differ between these datasets. These findings motivated researchers to consider lingual and social contexts in detection techniques. In Arabic spam tweets, the authors of [26] extracted text features using skip-grams and continuous bag of words (CBOW) to train ML classifiers and reached 87.3% accuracy with skip-grams learned using a corpus collected from Twitter. Another study examined tweet sentiment in conjunction with behavioral features to investigate the likelihood of a tweet being spam [27]. They found that the effectiveness of commenting behavior exceeds that of liking behavior in the classification process, whereas message sentiment is not a useful factor. Other researchers classified spam tweets in the Gulf Arabic dialect using content-based features with ML classifiers and attained 86% accuracy. Moreover, they stated that URL safety is insufficient in detecting Arabic spam and that the safety of the used words contributes more to the detection process [7]. The authors of [47] focused on spam in Saudi Arabia. They generated eight datasets, each collected from specific hashtags, which fall under three topics: national, health, and politics. After preprocessing, N-grams and Word2Vec embeddings were used for text feature extraction. These features were used to train different ML classifiers for the eight datasets separately, showing high performance using RF with N-grams. Moreover, ML classifiers were used in [48] with word embeddings and optimized by an augmentation technique that is based on Aravec's similarity. Then, these embeddings were combined with different features from three categories: content, user, and interaction. Hence, [10] focused on detecting advertisement spam in Arabic tweets by testing two methods: word embeddings with an ML classifier and fine-tuning AraBERT, a pre-trained Arabic language model. The latter achieved high accuracy compared with the former method. Moreover, the n-gram embeddings performed better than the Mazajak embeddings in the classification process. Another study [49] distinguishes between three types of spam: promotional, phishing, and spam. The last two categories are characterized based on the used URL: phishing if the URL directs to a trading website, and spam if it links to external websites. In their model, text features were extracted using Aravec for word embeddings in one model, whereas a character-level convolutional network was used for character embeddings in the other model. They integrated CNN for feature extraction with LSTM for spam classification. Their experiments demonstrated higher classification accuracy using word embeddings compared to character embeddings. In addition to regular text, [50] focused on classifying spam text within images. This was completed using an efficient and accurate scene text detector (EAST) and a convolutional recurrent neural network (CRNN) for text extraction .Then, the detection is performed by conducting comparisons between used words and two lists built by the authors: blacklist and whitelist. This method requires a big dataset in one language to avoid misclassification.

Moving to account-level detection, the authors of [25] classified Arabic spammers with 92.6% accuracy using 14 behavioral and content features selected by information gain (IG) and chi-squared feature selection methods. In addition to their previous finding that classification features are socially dependent, they observed how spammers employed different evasion techniques that may diminish the effectiveness of detection features over time. Similarly, the authors of [51] selected 16 features from a set of content- and user-based features and calculated different statistics for a subset of these features, including (total, minimum, maximum, and average). The rank of these features was determined using mean decrease Gini and subsequently reduced through the application of recursive feature elimination (RFE). Starting with about 47 features, more than 90% accuracy was achieved with 16 selected features. Comparing ML classifiers and deep neural networks (DNNs), the authors of [52] compared the two methods using text features. The features were extracted using N-grams (uni-gram, bi-gram, and char-gram) in the former and GloVe and fastText in the latter. DNN and GloVe outperformed other techniques in ML and DL models, respectively. Adapting the methodology used in [37], the authors of [53] proposed combining text features with metadata from tweets and accounts. Text features were extracted using word embeddings to train the CNN model, whereas a simple neural network was trained using 12 statistical features. The final classification is defined by combining the outputs of the two networks. Their framework achieved 94.27% accuracy in classifying Arabic spammer accounts. In addition to their proposed model for classifying spam tweets, the authors of [49] proposed another model for detecting Arabic spammers. After scraping a subset of tweets from the account, the embeddings were extracted (character level and word level) and used to train a deep learning model (CNN and LSTM), concluding that 10 to 15 tweets are optimal for high-performance classification. On the other hand, the authors of [14] focused on detecting types of spammers, namely those publishing porn content. Accordingly, different text features were tested individually and in combination. These features included username, screen name, user description, or a single tweet message extracted from the account. Text features were extracted from these attributes using n-gram comparison between word-level and character-level. Then, a Support Vector Machine (SVM) was trained using different alternatives of features. Other experiments involve fine-tuning pretrained language models, including Multi-BERT and AraBERT. These experiments revealed that screen names are not informative in the detection model, even with preprocessing and normalizing the used text. Moreover, the performance was highly improved by adding a single tweet with the username and description.

A review of the existing literature reveals several limitations that require further investigation. First, most studies rely on a single criterion for spam collection, such as URLs, hashtags, account suspension, or predefined keywords. These methods

restrict the diversity of captured spamming behaviors and limit model generalizability. In addition, datasets are often collected within short time frames and annotated using automated or semi-automated techniques, which increases sample homogeneity and labeling inaccuracies.

Second, prior studies adopt a binary classification model, typically distinguishing only between spam and non-spam messages, focusing on narrow categories such as URLs, advertisements, or pornographic content. Such approaches fail to capture the full spectrum of spamming behaviors, limiting the generalizability and robustness of the resulting models.

Third, previous studies often consider either the tweet level or the account level in isolation, operating under the assumption that spam tweets are produced by spammer accounts and that non-spam tweets are generated only by legitimate accounts. This assumption overlooks more scenarios, such as legitimate accounts posting occasional spam messages or spammer accounts generating legitimate content to evade detection techniques.

Finally, feature-based approaches often require extensive and continuously available user, content or network data, which may not always be accessible due to privacy restrictions, platform limitations, or availability.

III. Materials and Methods

A. Dataset

Data was gathered utilizing the Twitter Application Programming Interface (API) across several time intervals from 2018 to 2023. We employed a comprehensive array of keywords, encompassing user mentions, hashtags, and suspicious terms, to effectively identify diverse spamming targets. The data was manually annotated to classify user profiles as either legitimate or spammer, with spam tweets categorized into seven distinct spamming targets: commercial advertisements, medical advertisements, illegal services, sorcery content, pornographic content, monetary requests, and other irrelevant messages (TABLE I.). The annotation process was conducted manually by two annotators, with an auditor responsible for assigning the final labels. Both annotators are native Arabic speakers with expertise in computer information systems and were provided with detailed annotation guidelines to ensure accurate labeling of spam tweets and accounts. Inter-annotator consistency was evaluated using Cohen's Kappa (κ), yielding a value of $\kappa = 0.92$ at the tweet level and $\kappa = 0.9$ at the account level. Three versions were derived from this dataset as follows:

- DS_Multi_8: This version of tweet labels encompasses all seven categories of spam targets (Fig. 1), as well as the ham class.
- DS_Multi_4: In this version, the seven spam categories have been consolidated into three abstract categories: promoting spam (commercial advertisements, medical advertisements, and illegal services), unethical spam (sorcery content and pornographic content), and irrelevant spam (monetary requests and irrelevant content).
- DS_Binary: This version includes a binary label for tweets, denoting whether they are classified as spam or not.

The dataset comprises more than 15,000 records, with approximately 54% being spam tweets and the remaining 46% being non-spam tweets. As previously mentioned, different keyword types were used to collect spam tweets, with varying effectiveness. A variety of hashtags indicating Arabic country names were used. Of tweets using these hashtags, 59% were ham messages, and 41% were spam messages. The majority of tweets using sports-related hashtags were spam (68% vs. 32% ham). Trending hashtags were particularly targeted by spammers, with only 28% of tweets being ham and 72% being spam messages. User mentions were also analyzed, particularly replies to various Arabic news accounts (@AJABreaking, @AJArabic, @AlArabiya_Brk, @alqabas, @SaudiNews50, and @Sabqorg); 77% of these tweets were ham messages, while 23% tweets were spam messages. Finally, spam words associated with different spam types were examined. Only 5% of these tweets were ham messages, while 95% were spam.

At the account level, the dataset comprised 5,620 unique user accounts. Of these, 55% were identified as legitimate and active during the annotation process, while 36% were classified as active spammers. In addition, 4% of the accounts were suspended, and 5% had been deleted at the time of annotation.

TABLE I. Spam Tweet Categories

Category	Definition
Commercial Advertisement	Tweets promoting products or services other than verified account advertisements.
Illegal Medicine Advertisement	Tweets promoting medical products or services (e.g., weight loss products, sexual products).
Illegal Service	Tweets promoting illegal services, including those prohibited by the government, or inflating OSN metrics (e.g., number of followers, retweets, verification).
Money Request	Tweets asking for monetary help.
Sorcery Service	Tweets promoting sorcery services.
Pornographic Content	Tweets with pornographic content, including text or media.
Irrelevant Content	Tweets using hashtags or mentions unrelated to the tweet topic or tweets with only hashtags, mentions, or media.

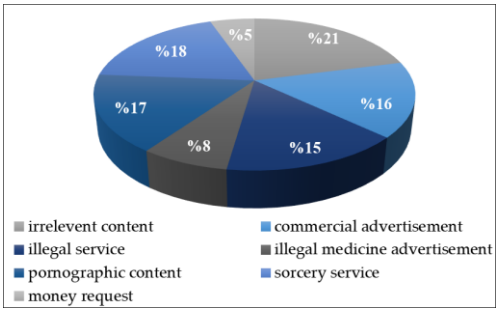


Fig. 1. Spam tweets per spam category.

B. AraSpam: Spam Detection Model Based on Multitask Learning

The AraSpam model is built to detect both spam tweets and accounts in Arabic Twitter. The model begins by extracting a

textual representation from the preprocessed tweet text and the profile screen name. The two extracted embeddings were subsequently stacked, supplemented by an oversampling technique, and then input into the multitask deep neural network. Two deep multitask neural network architectures were employed: hard parameter sharing and soft parameter sharing.

1) Text preprocessing: The preceding phase of embedding extraction from tweet text is preprocessing. The text was preprocessed with AraBERT Preprocessor following its default settings:

- Keeping emojis;
- Removing "html" markup;
- Replacing email, URL, and user mentions with specific tokens as follows:
- Replacing URL with [رابط], which means [link];
- Replacing user mention with [مستخدم], which means [user];
- Replacing email with [بريد], which means [email];
- Removing the repetition of more than two non-digit characters, e.g., (سلاااام) will be converted to (سلام);
- Stripping Tatweel (ـ), e.g., (محمـــد) will be converted to (محمد);
- Removing diacritics: Dammah (ُ), Tanween Aldam (ٌ), Kasra (ِ). Tanween Alkasr (ٍ), Fatha (َ), Tanween Alfath (ً), Skoun (ْ), and Shaddah (ّ);
- Replacing slash (/) with dash (-);
- Mapping Hindi numbers (١ ٢ ٣) to Arabic numbers (1 2 3);
- Inserting white space.

2) Text features extraction: Bidirectional Encoder Representations from Transformers (BERTs) is a transformer-based model that acquires linguistic representations by utilizing extensive text corpora through unsupervised methods such as masked language modeling (MLM) [54]. It can be fine-tuned to solve different NLP tasks, such as sentiment analysis and question answering. The model's input embeddings are the result of integrating three types of embeddings: token embeddings, segmentation embeddings, and position embeddings. Another version of BERT, trained on a Wikipedia corpus comprising 104 languages, is referred to as multilingual BERT (mBERT) [55]. It possesses advantages such as the tokenization method that can handle out-of-vocabulary words and its capacity to capture cross-lingual representation. An Arabic adaptation of the BERT model is AraBERT [56], which is trained using a large Arabic corpus and comes in different versions that differ based on the type and size of the training datasets, as shown in TABLE II. The original AraBERT dataset was crawled from Wikipedia and multiple Arabic corpora, including news articles from various Arab countries, covering a broad spectrum of topics written in Modern Standard Arabic (MSA). The latest versions incorporate tweets in diverse Arabic dialects, including emojis.

Sentence Transformers are a distinct category of transformer-based models designed to provide embeddings for sentences, thereby enabling effective evaluation of their semantics. Unlike traditional transformer models like BERT, which generate contextualized embeddings for individual tokens, Sentence Transformers construct a fixed-length vector representation for entire sentences. Sentence-BERT (SBERT) [57] and Language-agnostic BERT Sentence Embedding (LaBSE) [58] are transformer-based models built to provide embeddings for sentences or text. The last one was expanded to accommodate cross-lingual tasks, having been trained in 109 languages. Another sentence transformer mode supporting multilingual is E5 [59]. It generates dense, high-dimensional embeddings in which semantically similar inputs, regardless of language, are situated closer together in the embedding space. Accordingly, these models are highly effective in capturing semantic similarity, information retrieval, translation, and a range of other applications.

TABLE II. ARABERT MODEL VERSIONS [60]

Model	Pre-Segmentation (Farasa)	Dataset (Sentences/Size/nWords)
AraBERTv1-base	Yes	77M / 23GB / 2.7B
AraBERTv0.1-base	No	77M / 23GB / 2.7B
AraBERTv2-base	Yes	200M / 77GB / 8.6B
AraBERTv2-large	Yes	200M / 77GB / 8.6B
AraBERTv0.2-base	No	200M / 77GB / 8.6B
AraBERTv0.2-large	No	200M / 77GB / 8.6B
AraBERTv0.2-Twitter-base	No	AraBERTv0.2 dataset + 60M Multi-Dialect Tweets
AraBERTv0.2-Twitter-large	No	AraBERTv0.2 dataset + 60M Multi-Dialect Tweets

AraSpam utilizes word embeddings derived from two sources: preprocessed tweet text and user screennames, comparing between word- and sentence-level transformers as follows:

- Word level: extracting word embedding from preprocessed tweet text via AraBERTv0.2-Twitter-large and user screen names through mBERT.
- Sentence level: Three different alternative models were used for extracting sentence-level embeddings for both tweet text and account screen names, including SBERT, LaBSE, and E5.

3) Data resampling: To address the class imbalance problem, augmentation strategies involve either oversampling or undersampling. These strategies primarily aim to mitigate bias in the learning process against the majority class. The undersampling technique involves eliminating instances from the majority class, whereas oversampling approaches aim to augment the samples in the minority class, either randomly or through synthetic methods. One of these is the adaptive synthetic (ADASYN) sampling approach. It aims to account for

the more challenging samples that are identified throughout the augmentation process [61].

DS_Multi_8 and DS_Multi_4 exhibit an imbalance among the various spam tweet classifications. The ADASYN augmentation approach was employed to address the dataset imbalance.

4) Multitask deep neural network: A DNN is an ML approach that can process raw data, such as text, and derive the necessary representations for classification tasks. The DNN model comprises an artificial neural network with multiple hidden layers (Fig. 2). Every neuron in a layer takes inputs, assigns weights and bias, processes the sum through an activation function, and transmits the output to the subsequent layer [62]-[64], as in Eq. (1):

$$z = wx + b \tag{1}$$

where, z is the output, w is the weight, x is the input matrix, and b is the bias. The activation function can be one of several functions, mainly Sigmoid [Eq. (2)], Rectified Linear Unit (ReLU) [Eq. (3)], and Softmax [Eq. (4)], for multi-classification problems. The equations of these functions are as follows:

$$\text{Sigmoid: } \sigma(z) = \frac{1}{1+e^{-z}} \tag{2}$$

$$\text{ReLU: } f(z) = \max(0, z) \tag{3}$$

$$\text{Softmax: } \sigma(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}} \tag{4}$$

Finally, the loss function [Eq. (5)] calculates the difference between the predicted value ($\hat{y}_i$) and the actual value (y_i) as follows:

$$L = -\sum_i y_i \log(\hat{y}_i) \tag{5}$$

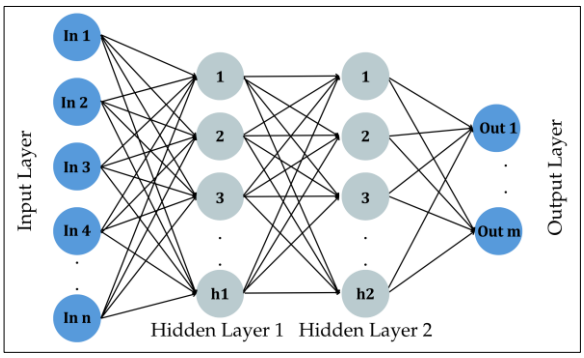


Fig. 2. General DNN structure.

Multitask learning (MTL) is a novel learning paradigm that leverages human learning to enable computers to learn multiple related tasks, thereby enhancing generalization performance [65]. At the beginning of MTL, the idea was motivated by the need to address the data scarcity problems that arise when different tasks have insufficiently labeled data for training. MTL combines these data to increase the model accuracy [66]. Formally, MTL is defined as "Given m learning tasks $\{\mathcal{T}\}_{i=1}^{m}$, where all the tasks or a subset of them are related, multi-task earning aims to learn the m tasks together to improve the learning of a model for each task $\mathcal{T}$i by using the knowledge contained in all or some of the other tasks [66]".

MTL tasks can be homogeneous, meaning all tasks share the same learning type, such as classification, regression, clustering, or others. Conversely, it could be heterogeneous if the learning tasks differ [67]. Joining the learning process between these tasks can be accomplished using different approaches, including feature learning, low-rank decomposition, task relationship learning, and task clustering. These approaches can be applied to tasks involving various learning types, including supervised, semi-supervised, unsupervised, reinforcement, online learning, and multi-view learning [66][68][69]. On the other hand, when using deep MTL, different methods of task concatenation can be employed, such as hard and soft parameter sharing, as shown in Fig. 3. The different tasks share the hidden layers in hard parameter sharing, while a constraint is added to encourage similarities among related parameters.

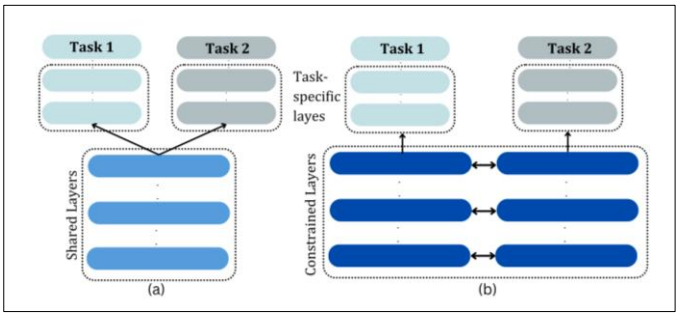


Fig. 3. Deep multitask learning approaches: (a) Hard parameter sharing, and (b) Soft parameter sharing.

a) Hard parameter sharing approach: The input layer functions as an embedding layer, as outlined in the preceding section for the tweet classification task. The identical layer is used for the account classification task after it is combined with the output of the tweet classification task layers. The tasks share certain hidden levels, whereas each task possesses additional task-specific layers, as shown in Fig. 4.

b) Soft parameter sharing approach: This architecture maintains the same input layer as the previous one, but the organization of the hidden layers differ. This method incorporates distinct hidden layers for each task, as illustrated in Fig. 5.

C. Experimental Setup

The experiments were conducted using Keras, an API integrated within the TensorFlow backend for implementing deep learning models. Consequently, the Keras tuner was utilized with 100 epochs to identify the optimal hyperparameters, including the number of units in the dense layer, dropout rate, and learning rate. Moreover, the Adam optimizer, L2 regularization (with a coefficient of 0.01), batch normalization, ReLU activation function, and cross-entropy loss function were used. Furthermore, transformer-based models were imported from the HuggingFace Transformers library. The experiments were conducted using Google Colab, equipped with high RAM and robust GPUs.

D. Performance Measures

The models' performances were measured using accuracy [Eq. (6)], macro F1-score [Eq. (7)], precision [Eq. (8)], and recall [Eq. (9)] as follows:

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{True Positives} + \text{True Negatives} + \text{False Positives} + \text{False Negatives}} \quad (6)$$

$$\text{F1} - \text{Score} = \frac{\text{False Positives}}{\text{True Positives} + \text{False Negatives}} \quad (7)$$

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positive}} \quad (8)$$

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \quad (9)$$

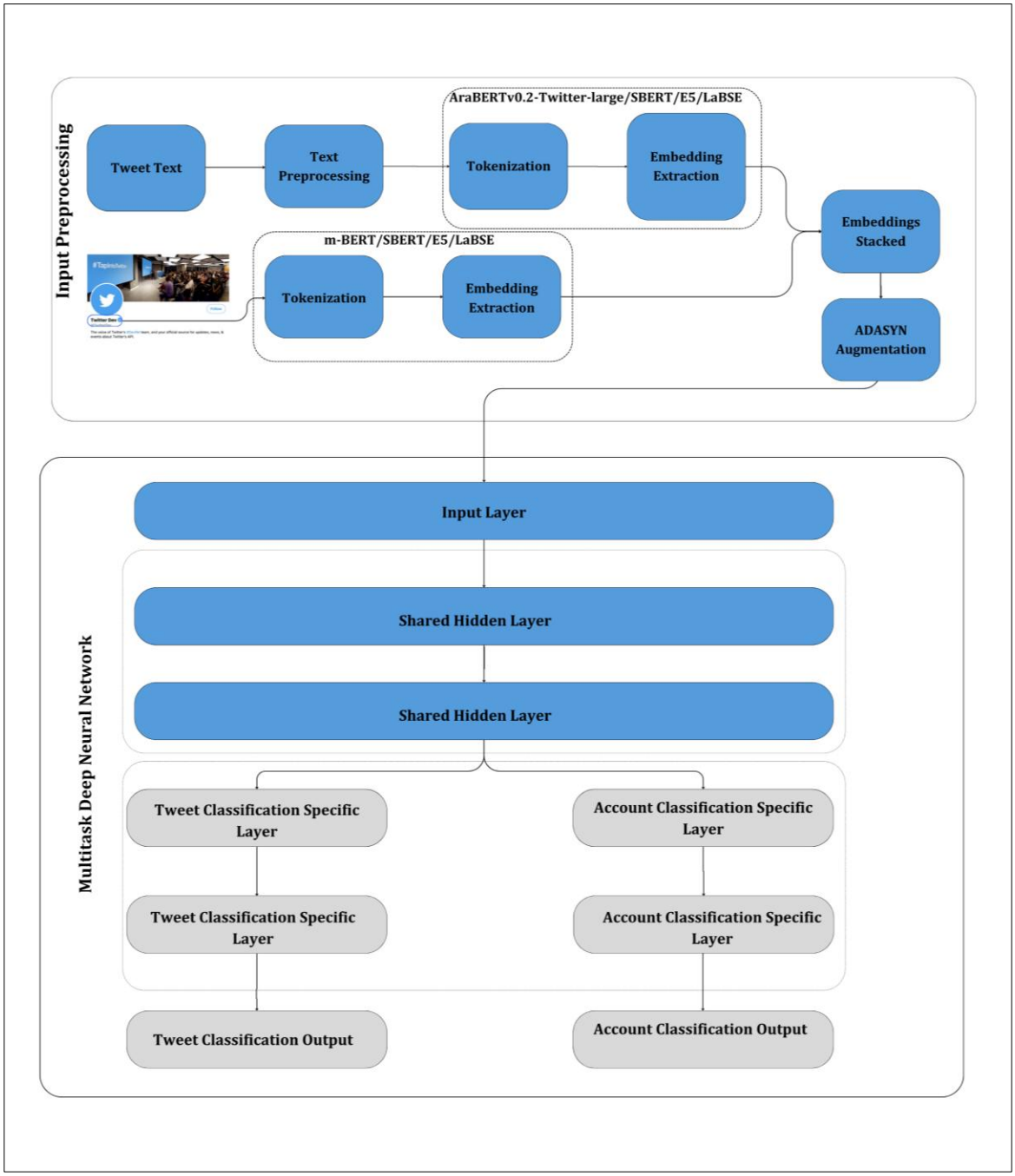


Fig. 4. AraSpam: Arabic spam detection model based on hard parameter sharing.

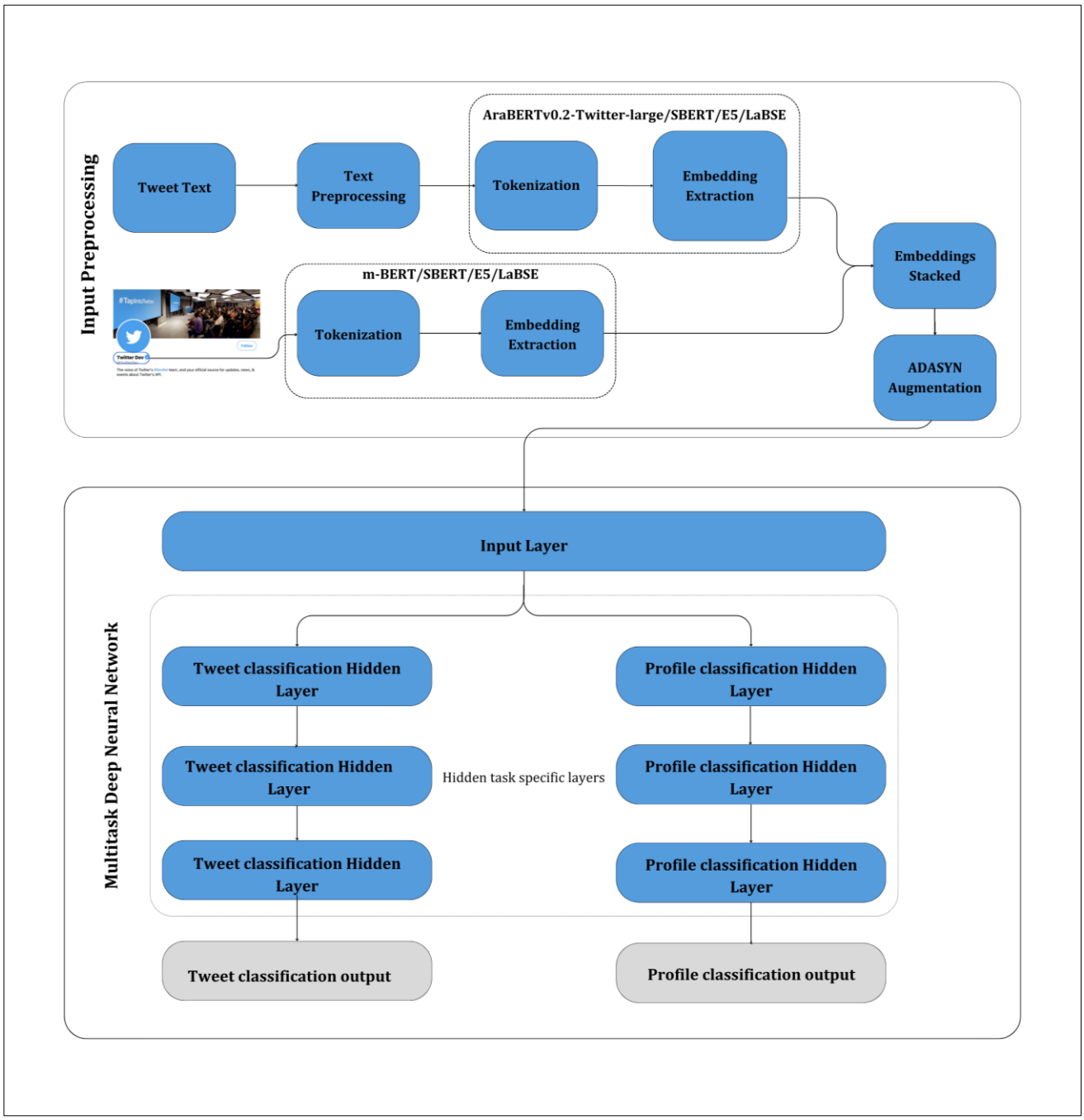


Fig. 5. AraSpam: Arabic spam detection model based on soft parameter sharing.

E. Model Validation

Various prior studies indicated that integrating text features with metadata enhances the accuracy of spam detection models [37][53]. AraSpam aims to achieve superior detection efficiency at both tweet and account levels with minimal prerequisites by eliminating the need to obtain metadata from Twitter. The model was validated by assessing and comparing the performance of three distinct inputs as follows:

- Spam detection metadata.
- Input 1: Tweet text embedding.
- Input 2: Tweet text embedding + screenname embedding.

Input 3: Tweet text embedding + screenname embedding + metadata (profile and content numerical features), as defined in TABLE III.

TABLE III. SPAM DETECTION METADATA

Feature name	Description
word_count	Number of words in the tweet
word_count_pre	Number of words in the preprocessed tweet
word_count_pre/ word_count	Ratio of words in the preprocessed tweet to the original tweet
char_count	Number of characters in the tweet
char_count_pre	Number of characters in the preprocessed tweet
char_count_pre/ char_count	Ratio of characters in the preprocessed tweet to the original tweet
semantic_similarity	Cosine similarity between tweet text and hashtag keywords embeddings
has_media	The presence of an image or video in a tweet
URLs_count	Number of URL(s) in tweet
mention_count	Number of users mentioned in the tweet
tags_count	Number of hashtags in a tweet
emoji_count	Number of emojis in the tweet
rt_count	How many times the tweet was retweeted
source	The software used to publish the tweet
fav_count	How many times the tweet was favorited
age	Number of days from the account creation date to the tweet publishing date
followers	Number of accounts following the user
friends	Number of accounts followed by the user
followers/friends	Ratio of followers to friends
reputation	Ratio of the number of followers to the total number of followers and friends
tweets_count	Number of tweets published by the user
tweets/day	Ratio of the number of tweets to account age
fav_list_count	Number of tweets favorited by the user
fav/day	Ratio of the number of favorites to account age
default_profile	If the user did not change the default profile bio
default_profile_image	If the user did not change the default profile image

IV. RESULTS AND DISCUSSION

This study focuses on building an efficient model to detect both Arabic spam messages and accounts on Twitter. Furthermore, it seeks to achieve high performance with minimal data retrieval requirements. The classification was conducted via text input, thereby avoiding the need to obtain metadata from Twitter. The features were derived from the text input using advanced transformer-based language models, comparing word-level and sentence-level transformers. The assessment was also conducted on two MTL paradigms: hard and soft parameter sharing. The proposed method was evaluated using three dataset versions: DS_Multi_8, DS_Multi_4, and DS_Binary. The performance measures achieved through these experiments are presented in TABLE IV. TABLE V. and TABLE VI. , and visualized in Fig. 6, Fig. 7, Fig. 8, Fig. 9, Fig. 10Fig. 11.

Across the three datasets, it is evident that an increase in tweet classes decreases tweet accuracy while enhancing account detection accuracy. In DS_Multi_8, the utilization of AraBERT embeddings showed superior efficacy in spam detection across both paradigms. Sentence transformer embeddings are more effective for account-level detection than for tweet-level detection. Following the reduction in spam classes from 7 to 3, tweet classification improved by approximately 1%, while account classification declined by roughly 2%. Moreover, AraBERT embeddings consistently provide superior performance. In the binary dataset, spam post detection improved, whereas account detection declined, and SBERT and LaBSE surpassed AraBERT at the account level by 1%.

The performance of hard and soft parameter sharing is nearly identical, exhibiting only minor variances. LaBSE and E5 exhibit superior performance using the hard parameter paradigm compared to the soft parameter sharing approach.

The model input was validated by comparing it with three inputs: Input 1, Input 2, and Input 3. The comparison of these inputs is shown in TABLE VII. . In DS_Binary, the experiment demonstrated consistent performance regardless of the input utilized. This conclusion considers Input 1 the optimal selection for this task, as the additional knowledge does not influence model performance. The performance of the multi-spam class datasets varies according to the chosen input. Input 2 achieved superior performance; however, utilizing Twitter text input (Input 1) resulted in a decrease in model performance of approximately 1%. Conversely, the incorporation of metadata (Input 2) has varied between degrading and maintaining the model's accuracy without alteration, determining that the effectual input is Input 2.

For the error analysis, the confusion matrix illustrates the classification performance of the model. Fig. 14 shows that the model achieved strong performance, particularly for sorcery content, pornographic content, money requests, and illegal medicine advertisements. However, a notable confusion arises between the irrelevant content and non-spam classes, with 39 irrelevant tweets mislabeled as non-spam and 21 non-spam tweets misclassified as irrelevant content. This suggests that the model struggles to distinguish ordinary tweets associated with irrelevant hashtags due to the similarity between the two categories. In addition, several commercial advertisement tweets were misclassified as illegal services (8 samples) or irrelevant content (13 samples), indicating that persuasive or promotional language sometimes overlaps with service-oriented or generic content. Such blurring may occur when advertisements avoid explicit commercial keywords. At the account level, Fig. 15 demonstrates excellent overall performance. The few misclassifications are likely attributable to legitimate accounts exhibiting spam-like behavior and, conversely, spammers attempting to mimic legitimate user activity.

Compared to prior research, the work by Mubarak et al. focused on a relatively narrow scope of spam detection, specifically targeting advertisements in [10] and accounts promoting pornographic content in [14]. Additionally, their first study limited the dataset to tweets targeting specific news accounts, thereby reducing generalizability. While Alharthi et al. [49] extended the classification to three spam categories, the model remained dependent on the presence of URLs, which restricted its applicability. Other studies, such as [51] and [52] , focused on spam detection based on specific spam-related keywords, while [53] and [25] primarily targeted spam associated with trending hashtags. In contrast, our proposed model addresses these limitations by generalizing spam classification across diverse categories without dependence on

predefined targets or URL-based heuristics. It was trained using a comprehensive dataset, collected by targeting hashtags, user mentions, and spam-related keywords.

Alharthi et al. [49] introduced two separate models—one for spam tweet detection and another for spam account detection—whereas our approach integrates both tasks into a single and unified model. Notably, their account-based model required a minimum of ten tweets per account to achieve reliable classification, while our model demonstrated high accuracy using only a single tweet. Alhassun et al. [53] proposed a framework that relies on both tweet content and extensive metadata, including premium features that require additional effort to extract. In contrast, our findings indicate that incorporating numerical features did not significantly enhance performance, simplifying the model and reducing dependency on complex metadata.

To the best of our knowledge, our model is the first to simultaneously address tweet-level and account-level spam detection in a single framework, highlighting different spamming categories.

TABLE IV. EXPERIMENTAL RESULTS USING DS_MULTI_8

		Hard Parameter Sharing				Soft Parameter Sharing			
task	**Metric**	**AraBERT**	**E5**	**SBERT**	**LaBSE**	**AraBERT**	**E5**	**SBERT**	**LaBSE**
Tweet	**Accuracy**	96	93	94	95	96	91	94	93
	F1-Score	95	93	93	94	96	89	93	93
	Precision	95	94	94	95	96	91	93	93
	Recall	96	92	93	93	95	87	93	92
Account	**Accuracy**	96	95	95	96	96	95	95	96
	F1-Score	96	95	95	96	96	95	95	96
	Precision	96	95	95	96	96	95	95	96
	Recall	96	95	95	96	96	95	95	96

TABLE V. EXPERIMENTAL RESULTS USING DS_MULTI_4

		Hard Parameter Sharing				Soft Parameter Sharing			
task	**Metric**	**AraBERT**	**E5**	**SBERT**	**LaBSE**	**AraBERT**	**E5**	**SBERT**	**LaBSE**
Tweet	**Accuracy**	97	94	95	96	97	92	95	95
	F1-Score	96	94	95	96	96	91	95	95
	Precision	97	94	95	96	97	93	95	95
	Recall	96	94	95	96	96	89	95	95
Account	**Accuracy**	94	93	93	94	94	94	93	94
	F1-Score	94	93	93	94	94	94	93	94
	Precision	94	93	93	94	94	94	93	94
	Recall	94	93	93	94	94	94	93	94

TABLE VI. EXPERIMENTAL RESULTS USING DS_BINARY

		Hard Parameter Sharing				Soft Parameter Sharing			
task	**Metric**	**AraBERT**	**E5**	**SBERT**	**LaBSE**	**AraBERT**	**E5**	**SBERT**	**LaBSE**
Tweet	**Accuracy**	98	96	96	96	98	95	96	96
	F1-Score	98	96	96	96	98	95	96	96
	Precision	98	96	96	96	98	95	96	96
	Recall	98	96	96	96	98	95	96	96
Account	**Accuracy**	91	90	92	92	90	90	91	92
	F1-Score	91	89	92	92	90	90	91	92
	Precision	91	91	92	92	90	90	91	92
	Recall	91	89	92	92	90	90	91	92

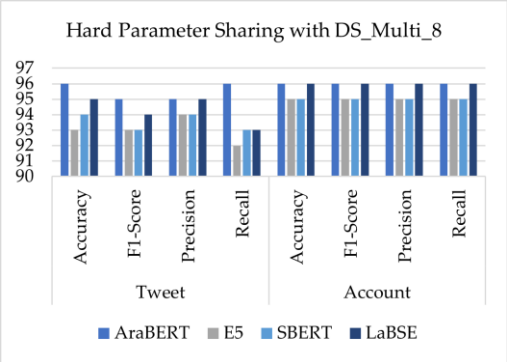


Fig. 6. Performance measures of the hard parameter sharing model with DS_Multi_8.

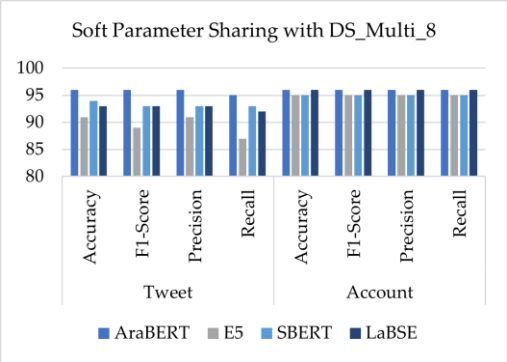


Fig. 7. Performance measures of the soft parameter sharing model with DS_Multi_8.

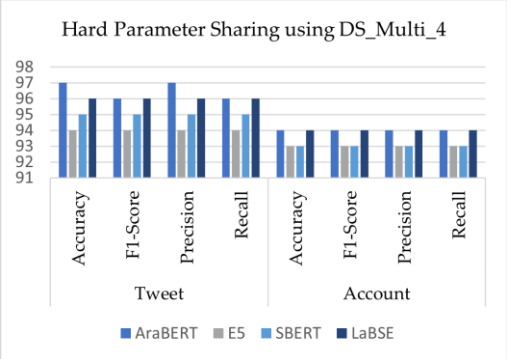


Fig. 8. Performance measures of the hard parameter sharing model with DS_Multi_4.

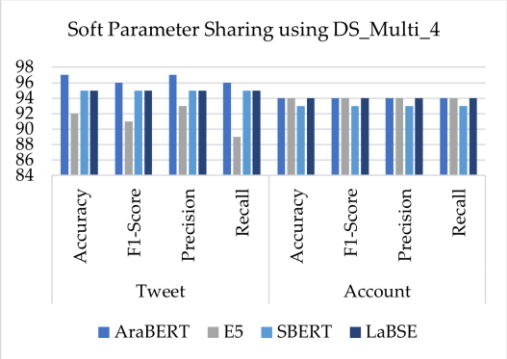


Fig. 9. Performance measures of the soft parameter sharing model with DS_Multi_4.

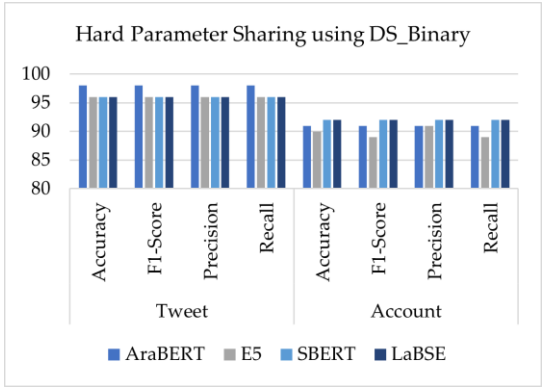


Fig. 10. Performance measures of the hard parameter sharing model with DS_Binary.

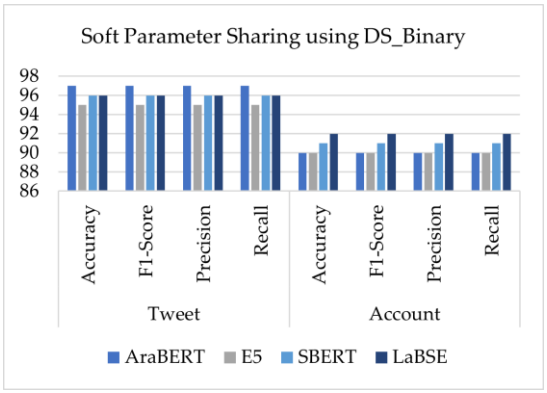


Fig. 11. Performance measures of the soft parameter sharing model with DS_Binary.

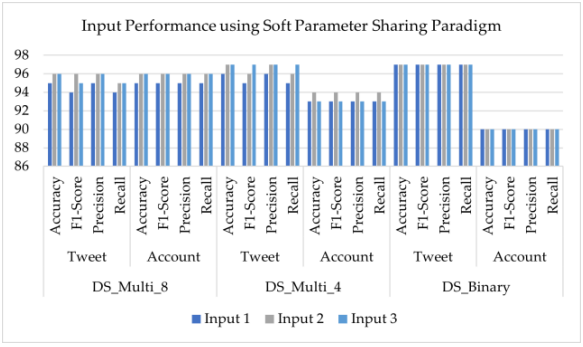


Fig. 12. Input performance using the hard parameter sharing paradigm.

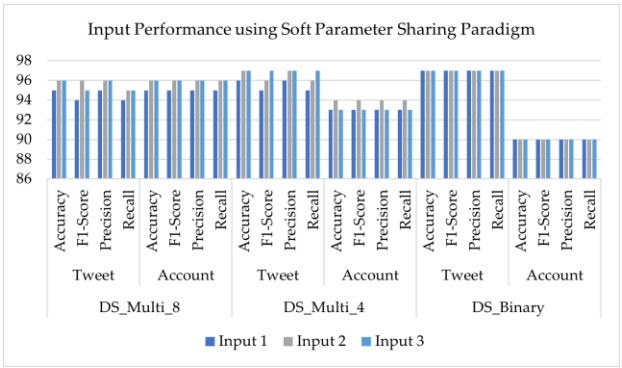


Fig. 13. Input performance using soft parameter sharing paradigm.

TABLE VII. PROPOSED INPUT PERFORMANCE

			Hard Parameter Sharing			Soft Parameter Sharing		
DS	**task**	**Metric**	**Input 1**	**Input 2**	**Input 3**	**Input 1**	**Input 2**	**Input 3**
DS_Multi_8	**Tweet**	**Accuracy**	95	96	95	95	96	96
		F1-Score	94	95	95	94	96	95
		Precision	95	95	95	95	96	96
		Recall	93	96	95	94	95	95
	Account	**Accuracy**	95	96	96	95	96	96
		F1-Score	95	96	96	95	96	96
		Precision	95	96	96	95	96	96
		Recall	95	96	96	95	96	96
DS_Multi_4	**Tweet**	**Accuracy**	95	97	96	96	97	97
		F1-Score	95	96	96	95	96	97
		Precision	95	97	96	96	97	97
		Recall	95	96	96	95	96	97
	Account	**Accuracy**	93	94	94	93	94	93
		F1-Score	93	94	94	93	94	93
		Precision	93	94	94	93	94	93
		Recall	93	94	94	93	94	93
DS_Binary	**Tweet**	**Accuracy**	98	98	98	97	98	97
		F1-Score	98	98	98	97	98	97
		Precision	98	98	98	97	98	97
		Recall	98	98	98	97	98	97
	Account	**Accuracy**	91	91	91	90	90	90
		F1-Score	91	91	91	90	90	90
		Precision	91	91	91	90	90	90
		Recall	91	91	91	90	90	90

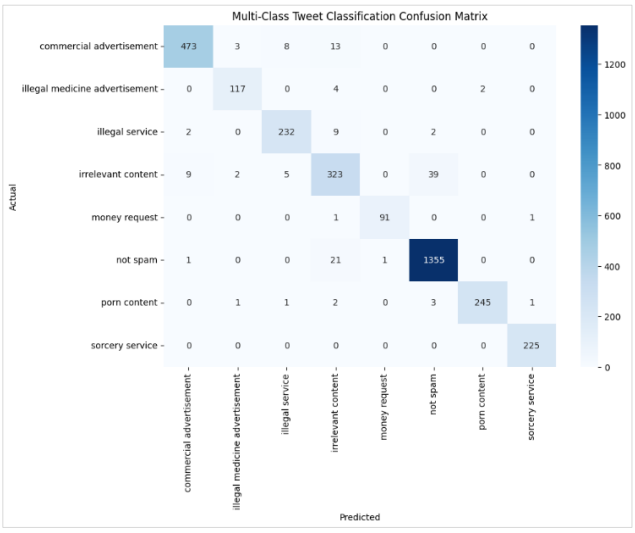


Fig. 14. Confusion matrix of spam tweet classification with DS_Multi_8.

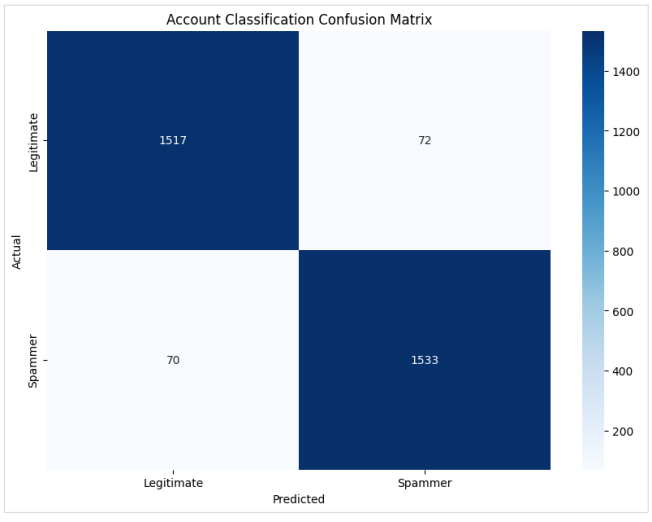


Fig. 15. Confusion matrix of spam account classification with DS_Multi_8.

V. CONCLUSION

This study introduced AraSpam, a multitask deep learning model designed to detect spam messages and spamming accounts in Arabic Twitter data, addressing limitations in prior models that focused on either content or user-level analysis in isolation. By leveraging transformer-based architectures such as AraBERT and mBERT for robust feature extraction, AraSpam achieved impressive performance, notably attaining a 96% accuracy rate on multi-class spam detection tasks.

The experimental evaluation across diverse datasets (DS_Multi_8, DS_Multi_4, and DS_Binary) highlighted the effectiveness of AraBERT in representing text and capturing semantics from tweet content. Notably, the results demonstrated that reducing the number of spam classes improved tweet-level classification while slightly compromising profile-level detection, reflecting the trade-off between granularity and generalization.

The study also explored different multitask learning paradigms, showing competitive results between hard and soft parameter sharing approaches, with LaBSE and E5 models performing particularly well under hard sharing. Furthermore, input validation experiments confirmed that Input 1, which avoids reliance on Twitter metadata, provides a practical balance between performance and feasibility.

Despite its strong results, AraSpam has room for enhancement. Future work should explore multimodal learning, incorporating image data from tweets to address visual spam content, which remains unprocessed in the current framework. Moreover, the dataset includes tweets with 280 characters maximum length and we are planning to enhance the dataset by including longer tweets.

In summary, AraSpam demonstrates the value of multitask learning and linguistically aware models in tackling the complex challenge of spam detection in Arabic social media.

REFERENCES

[1] E. Ferrara, "The history of digital spam," Commun. ACM, vol. 62, no. 8, pp. 82–91, 2019, doi: 10.1145/3299768.

[2] L. F. Cranor and B. A. LaMacchia, "Spam!," Commun. ACM, vol. 41, no. 8, pp. 74–83, 1998, doi: 10.1145/280324.280336.

[3] F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida, "Detecting spammers on twitter," in Collaboration, electronic messaging, anti-abuse and spam conference (CEAS), 2010, vol. 6, no. 2010, p. 12.

[4] X. H. Center. "Platform manipulation and spam policy." (accessed 14th January, 2025).

[5] A. H. Wang, "Don't follow me: Spam detection in twitter," in 2010 international conference on security and cryptography (SECRYPT), 2010: IEEE, pp. 1-10.

[6] S. Sedhai and A. Sun, "Hspam14: A collection of 14 million tweets for hashtag-oriented spam research," in Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval, 2015, pp. 223-232.

[7] D. Alorini and D. B. Rawat, "Automatic spam detection on gulf dialectical Arabic Tweets," in 2019 International Conference on Computing, Networking and Communications (ICNC), 2019: IEEE, pp. 448-452.

[8] H. S. Al-Khalifa, "On the analysis of twitter spam accounts in Saudi Arabia," International Journal of Technology Diffusion (IJTD), vol. 6, no. 1, pp. 46-60, 2015.

[9] N. Jindal and B. Liu, "Opinion spam and analysis," in Proceedings of the 2008 international conference on web search and data mining, 2008, pp. 219-230.

[10] H. Mubarak, A. Abdelali, S. Hassan, and K. Darwish, "Spam detection on arabic twitter," in International Conference on Social Informatics, 2020: Springer, pp. 237-251.

[11] K. S. Adewole, N. B. Anuar, A. Kamsin, K. D. Varathan, and S. A. Razak, "Malicious accounts: dark of the social networks," Journal of Network and Computer Applications, vol. 79, pp. 41-67, 2017.

[12] S. W. Liew, N. F. M. Sani, M. T. Abdullah, R. Yaakob, and M. Y. Sharum, "An effective security alert mechanism for real-time phishing tweet detection on Twitter," Computers & Security, vol. 83, pp. 201-207, 2019.

[13] A. T. Kabakus and R. Kara, "A survey of spam detection methods on twitter," International Journal of Advanced Computer Science and Applications, vol. 8, no. 3, pp. 29-38, 2017.

[14] H. Mubarak, S. Hassan, and A. Abdelali, "Adult content detection on arabic twitter: Analysis and experiments," in Proceedings of the Sixth Arabic Natural Language Processing Workshop, 2021, pp. 136-144.

[15] M. H. Arif, J. Li, M. Iqbal, and K. Liu, "Sentiment analysis and spam detection in short informal text using learning classifier systems," Soft Computing, vol. 22, no. 21, pp. 7281-7291, 2018.

[16] S. B. Abkenar, M. H. Kashani, M. Akbari, and E. Mahdipour, "Twitter spam detection: A systematic review," arXiv preprint arXiv:2011.14754, 2020.

[17] T. Wu, S. Wen, Y. Xiang, and W. Zhou, "Twitter spam detection: Survey of new approaches and comparative study," Computers & Security, vol. 76, pp. 265-284, 2018/07/01/ 2018, doi: https://doi.org/10.1016/j.cose.2017.11.013.

[18] T. Wu, S. Liu, J. Zhang, and Y. Xiang, "Twitter spam detection based on deep learning," in Proceedings of the australasian computer science week multiconference, 2017, pp. 1-8.

[19] S. Rao, A. K. Verma, and T. Bhatia, "A Review on Social Spam Detection: Challenges, Open Issues, and Future Directions," Expert Systems with Applications, p. 115742, 2021.

[20] M. Chakraborty, S. Pal, R. Pramanik, and C. R. Chowdary, "Recent developments in social spam detection and combating techniques: A survey," Information Processing & Management, vol. 52, no. 6, pp. 1053-1073, 2016.

[21] F. Persia and D. D'Auria, "A survey of online social networks: challenges and opportunities," in 2017 IEEE International Conference on Information Reuse and Integration (IRI), 2017: IEEE, pp. 614-620.

[22] H. Afzal and K. Mehmood, "Spam filtering of bi-lingual tweets using machine learning," in 2016 18th International Conference on Advanced Communication Technology (ICACT), 2016: IEEE, pp. 710-714.

[23] A. M. Al-Zoubi, J. f. Alqatawna, H. Faris, and M. A. Hassonah, "Spam profiles detection on social networks using computational intelligence methods: The effect of the lingual context," Journal of Information Science, p. 0165551519861599, 2019.

[24] A.-Z. Ala'M, H. Faris, J. f. Alqatawna, and M. A. Hassonah, "Evolving support vector machines using whale optimization algorithm for spam profiles detection on online social networks in different lingual contexts," Knowledge-Based Systems, vol. 153, pp. 91-104, 2018.

[25] N. El-Mawass and S. Alaboodi, "Detecting Arabic spammers and content polluters on Twitter," in 2016 Sixth International Conference on Digital Information Processing and Communications (ICDIPC), 2016: IEEE, pp. 53-58.

[26] S. Al-Azani and E.-S. M. El-Alfy, "Detection of arabic spam tweets using word embedding and machine learning," in 2018 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT), 2018: IEEE, pp. 1-5.

[27] M. M. Alsulami and A. Y. Al-Aama, "SentiFilter: A Personalized Filtering Model for Arabic Semi-spam Content Based on Sentimental and Behavioral Analysis," Int. J. Adv. Comput. Sci. Appl, vol. 11, 2020.

[28] H. Gupta, M. S. Jamal, S. Madisetty, and M. S. Desarkar, "A framework for real-time spam detection in Twitter," in 2018 10th International Conference on Communication Systems & Networks (COMSNETS), 2018: IEEE, pp. 380-383.

[29] X. Ban, C. Chen, S. Liu, Y. Wang, and J. Zhang, "Deep-learnt features for Twitter spam detection," in 2018 International Symposium on Security and Privacy in Social Networks and Big Data (SocialSec), 2018: IEEE, pp. 208-212.

[30] M. Neha and M. S. Nair, "A novel twitter spam detection technique by integrating inception network with attention based lstm," in 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), 2021: IEEE, pp. 1009-1014.

[31] K. Maithili et al., "An Effective Twitter Spam Detection Model using Multiple Hidden Layers Extreme Learning Machine," International Journal of Intelligent Systems and Applications in Engineering, vol. 12, no. 1s, pp. 01-09, 2024.

[32] A. Ghourabi and M. Alohaly, "Enhancing spam message classification and detection using transformer-based embedding and ensemble learning," Sensors, vol. 23, no. 8, p. 3861, 2023.

[33] M. S. Karakaşlı, M. A. Aydin, S. Yarkan, and A. Boyaci, "Dynamic feature selection for spam detection in Twitter," in International Telecommunications Conference, 2019: Springer, pp. 239-250.

[34] I. Inuwa-Dutse, M. Liptrott, and I. Korkontzelos, "Detection of spam-posting accounts on Twitter," Neurocomputing, vol. 315, pp. 496-511, 2018/11/13/ 2018, doi: https://doi.org/10.1016/j.neucom.2018.07.044.

[35] F. Concone, G. L. Re, M. Morana, and C. Ruocco, "Twitter Spam Account Detection by Effective Labeling," in ITASEC, 2019.

[36] K. S. Adewole, N. B. Anuar, A. Kamsin, and A. K. Sangaiah, "SMSAD: a framework for spam message and spam account detection," Multimedia Tools and Applications, vol. 78, no. 4, pp. 3925-3960, 2019/02/01 2019, doi: 10.1007/s11042-017-5018-x.

[37] Z. Alom, B. Carminati, and E. Ferrari, "A deep learning model for Twitter spam detection," Online Social Networks and Media, vol. 18, p. 100079, 2020.

[38] B. Abu-Salih et al., "An intelligent system for multi-topic social spam detection in microblogging," Journal of Information Science, vol. 50, no. 6, pp. 1471-1498, 2024.

[39] M. Washha, A. Qaroush, and F. Sedes, "Leveraging time for spammers detection on Twitter," presented at the Proceedings of the 8th International Conference on Management of Digital EcoSystems, Biarritz, France, 2016. [Online]. Available: https://doi.org/10.1145/3012071.3012078.

[40] H. Tajalizadeh and R. Boostani, "A novel stream clustering framework for spam detection in Twitter," IEEE Transactions on Computational Social Systems, vol. 6, no. 3, pp. 525-534, 2019.

[41] F. Concone, G. L. Re, M. Morana, and C. Ruocco, "Assisted Labeling for Spam Account Detection on Twitter," in 2019 IEEE International Conference on Smart Computing (SMARTCOMP), 2019: IEEE, pp. 359-366.

[42] K. S. Adewole, T. Han, W. Wu, H. Song, and A. K. Sangaiah, "Twitter spam account detection based on clustering and classification methods," The Journal of Supercomputing, vol. 76, no. 7, pp. 4802-4837, 2020.

[43] R. Alharthi, A. Alhothali, and K. Moria, "Detecting and characterizing arab spammers campaigns in Twitter," Procedia Computer Science, vol. 163, pp. 248-256, 2019.

[44] P.-C. Chen, H.-M. Lee, H.-R. Tyan, J.-S. Wu, and T.-E. Wei, "Detecting spam on Twitter via message-passing based on retweet-relation," in International Conference on Technologies and Applications of Artificial Intelligence, 2014: Springer, pp. 56-65.

[45] S. Gupta, A. Khattar, A. Gogia, P. Kumaraguru, and T. Chakraborty, "Collective classification of spam campaigners on Twitter: A hierarchical meta-path based approach," in Proceedings of the 2018 World Wide Web Conference, 2018, pp. 529-538.

[46] D. Antonakaki, I. Polakis, E. Athanasopoulos, S. Ioannidis, and P. Fragopoulou, "Exploiting abused trending topics to identify spam campaigns in Twitter," Social Network Analysis and Mining, vol. 6, no. 1, p. 48, 2016/07/13 2016, doi: 10.1007/s13278-016-0354-9.

[47] A. Balfagih, V. Keselj, and S. Taylor, "N-gram and word2vec feature engineering approaches for spam recognition on some influential twitter topics in saudi arabia," in Proceedings of the 6th International Conference on Information System and Data Mining, 2022, pp. 101-107.

[48] A. M. Alkadri, A. Elkorany, and C. Ahmed, "Enhancing detection of Arabic social spam using data augmentation and machine learning," Applied Sciences, vol. 12, no. 22, p. 11388, 2022.

[49] R. Alharthi, A. Alhothali, and K. Moria, "A real-time deep-learning approach for filtering Arabic low-quality content and accounts on Twitter," Information Systems, vol. 99, p. 101740, 2021.

[50] N. Imam and V. Vassilakis, "Detecting spam images with embedded arabic text in twitter," in 2019 international conference on document analysis and recognition workshops (ICDARW), 2019, vol. 6: IEEE, pp. 1-6.

[51] A. R. Alharbi and A. Aljaedi, "Predicting rogue content and arabic spammers on twitter," Future Internet, vol. 11, no. 11, p. 229, 2019.

[52] S. Kaddoura, S. A. Alex, M. Itani, S. Henno, A. AlNashash, and D. J. Hemanth, "Arabic spam tweets classification using deep learning," Neural Computing and Applications, vol. 35, no. 23, pp. 17233-17246, 2023.

[53] A. S. Alhassun and M. A. Rassam, "A combined text-based and metadata-based deep-learning framework for the detection of spam accounts on the social media platform twitter," Processes, vol. 10, no. 3, p. 439, 2022.

[54] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, 2018.

[55] B. Muller, A. Anastasopoulos, B. Sagot, and D. Seddah, "When being unseen from mBERT is just the beginning: Handling new languages with multilingual language models," arXiv preprint arXiv:2010.12858, 2020.

[56] W. Antoun, F. Baly, and H. Hajj, "Arabert: Transformer-based model for arabic language understanding," arXiv preprint arXiv:2003.00104, 2020.

[57] N. Reimers, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," arXiv preprint arXiv:1908.10084, 2019.

[58] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, and W. Wang, "Language-agnostic BERT sentence embedding," arXiv preprint arXiv:2007.01852, 2020.

[59] L. Wang, N. Yang, X. Huang, L. Yang, R. Majumder, and F. Wei, "Multilingual e5 text embeddings: A technical report," arXiv preprint arXiv:2402.05672, 2024.

[60] W. Antoun and F. Baly. "AraBERTv0.2-Twitter." (accessed 1st March, 2025).

[61] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in 2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence), 2008: Ieee, pp. 1322-1328.

[62] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," nature, vol. 521, no. 7553, pp. 436-444, 2015.

[63] H. Larochelle, Y. Bengio, J. Louradour, and P. Lamblin, "Exploring strategies for training deep neural networks," Journal of machine learning research, vol. 10, no. 1, 2009.

[64] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "Efficient processing of deep neural networks: A tutorial and survey," Proceedings of the IEEE, vol. 105, no. 12, pp. 2295-2329, 2017.

[65] R. Caruana, "Multitask learning," Machine learning, vol. 28, no. 1, pp. 41-75, 1997.

[66] Y. Zhang and Q. Yang, "A survey on multi-task learning," IEEE Transactions on Knowledge and Data Engineering, 2021.

[67] X. Yang, S. Kim, and E. Xing, "Heterogeneous multitask learning with joint sparsity constraints," Advances in neural information processing systems, vol. 22, pp. 2151-2159, 2009.

[68] Y. Zhang and Q. Yang, "An overview of multi-task learning," National Science Review, vol. 5, no. 1, pp. 30-43, 2018.

[69] S. Ruder, "An overview of multi-task learning in deep neural networks," arXiv preprint arXiv:1706.05098, 2017.

Hierarchical Transformer Residual Model for Pneumonia Detection and Lesion Mapping

Anupama Prasanth
College of Computer Studies, University of Technology Bahrain, Salmabad, Kingdom of Bahrain

Abstract*—Pneumonia, a potentially fatal infection and a common disease-causing culprit among children and the elderly, still remains as a prevalent threat even after years of research on tackling it. Rapid and proper identification is crucial for timely treatment and improved results. While thoracic radiographs are widely employed in pneumonia diagnosis, real-world clinical assessment is frequently questioned by factors such as subtle radiographic patterns, overlapping symptoms, subjective manual judgement and dependency on expert radiologists. The study proposes a hybrid deep learning model integrating ResNet50 and the Swin Transformer, coupled with an auxiliary segmentation decoder to facilitate both classification and lesion localization in chest X-ray images. ResNet50 acts as the backbone for hierarchical spatial feature extraction, capturing fine-grained local textures indicative of pulmonary abnormalities, and the Swin Transformer serves as the global attention-driven feature aggregator. The shifted window mechanism of the Swin Transformer maintains spatial hierarchy while facilitating effective contextual modelling. Global Average Pooling (GAP) and Multilayer Perceptron (MLP) form the classification head, yielding accurate predictions in classifying the images, while the segmentation decoder utilizes multiscale features to generate pixel-wise masks for pneumonia lesion regions. The model outperformed conventional methods with 98.4% classification accuracy, 98.2% precision, 99.2% recall and an F1-score of 98.7% with a 0.88 Dice Coefficient in segmentation. These results reflect the hybrid architecture's superior performance and its dual capacity for diagnostic prediction and lesion interpretability. The proposed model demonstrates promising results for deployment in real-world clinical workflows, especially in resource-constrained or high-patient-load environments.

Keywords*—Pneumonia detection; lesion segmentation; chest X-ray; ResNet50; swin transformer; global average pooling

I. Introduction

Pneumonia is a potentially life-threatening pulmonary infection causing inflammation of the alveoli, the tiny air sacs within the lungs, filling up with fluid or pus. Affecting over 150 million individuals and causing around 2.5 million deaths every year, it is the primary cause of infectious disease deaths among children under five years old globally, responsible for 70,000 deaths per year, a terrifying 14% of deaths [1]. A number of pathogens, including bacteria, viruses and fungi, could act as the agents of the fatal disease. Streptococcus pneumoniae remains the most common bacterial agent, and respiratory viruses such as influenza and SARS-CoV-2 also contribute significantly to disease incidence [2]. Prolonged coughing, chest pain, fever, chills and respiratory difficulties are common signs of pneumonia, and in extreme cases, additional symptoms like confusion, cyanosis and fast breathing can also occur. Delayed or inaccurate diagnosis results in disease progression, complications or even death, highlighting the critical need for timely and reliable diagnostic methodologies [3].

Clinical examination, auscultation, laboratory tests and manual chest radiography analysis are the basic traditional pneumonia detection methods. Clinical examination and auscultation have limited sensitivity and delayed turnaround times, are frequently used only as a manual screening process, and that too with high error rates in early pneumonia manifestations [4]. Due to the unavailability of specific biomarkers, laboratory tests frequently fail to pinpoint pneumonia from other respiratory illnesses. Even though chest X-rays offer a non-invasive and accessible tool for visual confirmation of pneumonia, their interpretation is subjective, depending heavily on the expertise of a radiologist [5]. Misinterpretation or variability in radiographic evaluations usually leads to delayed diagnosis or misdiagnosis. Traditional image analysis techniques have limited capacity to discern subtle pathological patterns or overlapping pulmonary anomalies, especially in early or atypical Pneumonia infections [6].

Deep learning (DL) based models have displayed promising results in automated pneumonia detection methods from chest radiographs in their initial phases. Convolutional neural networks (CNNs), DenseNet and Vision Transformers (ViT) frameworks have demonstrated considerable success in enhancing classification accuracy and reducing diagnostic turnaround time [7]. But the lack of spatial interpretability, being computationally complex and costly, hinder real-time deployment of most of the models. Limited interpretability and high resource requirements, and skilled hardware operation requirements make them irrelevant in rural and resource constrained regions. Moreover, few integrate simultaneous segmentation of pathological regions, which is vital for localized analysis and clinical insight.

The study proposes a novel ResNet50-Swin Transformer hybrid model, addressing the diagnostic challenges by combining robust local feature extraction with global contextual representation, enabling both accurate classification and precise segmentation of pneumonia from chest X-rays. This dual-capability framework enhances diagnostic reliability while maintaining computational efficiency rendering it ideal for practical clinical deployment. To bridge the gap between theoretical accuracy and practical feasibility, the proposed model aims to design a model that not only performs robustly in controlled settings but is also scalable and reliable in high-patient-load environments such as rural clinics or emergency care centers lacking expert radiologists. By reducing reliance on

manual interpretation and enabling automatic lesion localization, the hybrid framework holds the potential to significantly reduce diagnostic delays in telemedicine setups and frontline screening programs. The lightweight architectural design further enables deployment on edge devices such as mobile radiology units and AI-enabled X-ray machines, making it especially useful in developing countries or disaster-struck zones. In real-world hospital settings, such a model could assist junior clinicians or non-specialist health workers in triaging suspected pneumonia cases more efficiently, improving both speed and consistency of care.

The core research question addressed in this study is whether a hybrid model integrating convolutional and transformer-based architectures with an integrated segmentation decoder enhances both classification accuracy and lesion-level interpretability in pneumonia detection from chest radiographic images. The key objectives of the study are as mentioned below:

- Design and implement a hybrid DL framework that leverages the spatial feature extraction capabilities of ResNet50 with contextual attention mechanisms of the Swin Transformer for binary classification of thoracic radiograph images into affected and healthy cases.
- Incorporate an integrated segmentation decoder within the hybrid model to generate pixel-wise lesion maps, enabling accurate localization of pneumonia-affected regions and thereby improving diagnostic interpretability and clinical decision-making.

The further sections of the research are structured as follows: Section II offers a comprehensive analysis of the latest advances in Pneumonia detection research and highlights the current research constraints. Section III outlines the proposed methodology. Experimental findings are presented in Section IV, accompanied by a comprehensive analysis of the model performance in Section V. Section VI concludes the study by encapsulating the principal results and emphasizing prospective areas for further research.

II. Related Works

Ali et al. [8] proposed a DL-based diagnostic framework to identify pneumonia from thoracic radiograph images. Six CNN architectures were implemented and trained on the dataset. EfficientNetV2L outperformed others with an accuracy of 94.02%, followed by VGG16 (91.66%), Xception (90.7%), InceptionResNetV2 (88.94%), ResNet50 (87.98%) and a baseline CNN (87.78%). The high performance of EfficientNetV2L was attributed to the compound scaling strategy and efficient use of network parameters, enabling the framework for fine-grained image classification tasks. However, the absence of region-wise interpretability restricted the clinical transparency of model predictions.

Shaikh et al. [9] proposed MDEV, an ensemble model for thoracic radiograph image classification into Pneumonia affected and healthy categories constructed by concatenating four deep transfer learning models: MobileNet, DenseNet-201, EfficientNet-B0 and VGG-16. Each component was fine-tuned and trained on the selected dataset and achieved an accuracy of 92.15% on evaluation. The MDEV model operated in two hierarchical levels: in the first level, learners independently extracted features from pre-processed thoracic radiograph images using distinct deep networks and in the second level, learning was performed by a meta-learner that integrated the outputs. The architectural complexity led to reduced interpretability and higher computational requirements, hampering the real-time deployment in low-resource or point-of-care environments.

Barhoom et al. [10] proposed a CNN framework for the Pneumonia identification and classification using thoracic radiograph images. The study utilized the hierarchical feature extraction capabilities of CNN to distinguish between normal and infected lungs and utilized a dataset of chest radiographs, focusing on three output classes: bacterial pneumonia, viral pneumonia and normal lungs. Among the tested architectures, VGG16 demonstrated superior performance, attributed to its deep structure and effective extraction of low- to high-level spatial features. The CNN architecture allowed for scalable feature learning and enabled accurate triaging of pneumonia subtypes, clinically valuable for treatment planning. The absence of cross-validation or external dataset benchmarking limited the model's generalizability to broader clinical populations and varying imaging conditions.

Wang et al. [11] suggested a pneumonia classification framework based on DenseNet, addressing the structural complexity and uneven gray-level distribution in chest X-ray images. The core architecture utilized DenseNet due to its ability to propagate learned features across all layers, supporting parameter efficiency and improved local feature learning. Squeeze and Excitation (SE) blocks, a channel attention mechanism, were integrated into the network to emphasize pneumonia-related information in the feature maps. The integration of attention modules and pooling adjustments contributed to a more discriminative lesion focus. Evaluated on the Chest X-ray 2017 dataset, the modified DenseNet achieved an accuracy of 92.8%. The study's precision and recall balance was flawed, indicating the need for further optimization for proper classification.

Mabrouk et al. [12] presented an ensemble learning (EL) approach for the computer-aided pneumonia classification utilizing lung radiograph images. The model integrated three pretrained architectures: DenseNet169, MobileNetV2 and ViT, fine-tuned on pediatric chest radiographs from the Chest X-ray Images (Pneumonia) dataset. Each model acted as a functional layer, independently extracting feature representations from the input images, and these features were passed through GAP for dimensionality reduction and subsequently combined to form the final predictive output. The proposed EL approach achieved 93.91% accuracy but was limited by the computational complexity, as the simultaneous use of three deep architectures increased inference time and resource consumption, hampering deployment in real-time or resource-limited clinical environments.

Ortiz-Toro et al. [13] explored the diagnostic potential of three textural image characterization techniques as input biomarkers for artificial intelligence (AI) models in pneumonia detection from thoracic radiograph images. Two datasets, Guangzhou Women and Children Medical Center pediatric

dataset (GWCMCx) and a composite dataset (Josep-NIH) including COVID-19 chest X-rays were utilized and three machine learning (ML) classifiers: Support Vector Machine (SVM), K-Nearest Neighbors (KNN) and Random Forest (RF), were employed in model evaluation. The study integrated handcrafted feature descriptors with conventional classifiers and demonstrated the utility of texture-based biomarkers in distinguishing pneumonia-affected lungs from normal cases with minimal preprocessing. On the GWCMCx dataset, the super pixel-based histon method yielded the highest accuracy at 91.3%, outperforming radiomics (83.3% accuracy) and fractal dimension (89.9%). On the Josep-NIH dataset, the fractal dimension achieved better accuracy, but the latter dataset used for evaluation was more biased, hampering the reliability.

Bhatt et al. [14] proposed an ensemble-based CNN framework for pneumonia diagnosis using lung radiographic images. The architecture comprised three parallel CNN models with distinct kernel sizes, each capturing features at varying spatial resolutions and the outputs were fused through a weighted ensemble mechanism that enabled dynamic thresholding, allowing clinicians to fine-tune diagnostic sensitivity based on clinical requirements. The framework maintained a lightweight footprint, and the ensemble increased feature diversity, robustness and allowed personalized classification thresholds for clinical flexibility with an accuracy of 84.12% and precision of 80.04%. The model incurred a higher computational cost than a single CNN, and the dataset size introduced a risk of overfitting.

Singh et al. [15] proposed an attention-aware CNN architecture for pneumonia detection in thoracic radiograph images. The study integrated channel and spatial attention modules within a deep neural network (DNN) to guide the learning process toward clinically relevant regions. A bottom-up and top-down feedforward attention mechanism was adopted, allowing both feedforward and feedback attention processes to coexist within each module. The dual-level attention provided enhanced feature discrimination and allowed the model to dynamically localize pneumonia-affected regions. The network achieved a 95.47% accuracy, outperforming both baseline CNN and a pretrained ResNet50 model. Comparative evaluation revealed that ResNet50 with integrated attention modules improved accuracy from 84.53% to 95.73%. The attention modules however, increased architectural complexity, resulting in extended training time and requiring more computational resources in deployment scenarios.

Ibrahim et al. [16] presented a DL framework employing a pretrained AlexNet model to segregate chest X-ray images across multiple diagnostic scenarios. By utilizing AlexNet's hierarchical feature extraction capabilities, the study handled both binary and multiclass classification tasks and demonstrated functional adaptability for a range of diagnostic applications in chest radiology. On evaluation through two-way, three-way and four-way classification tasks using publicly available datasets, the model achieved accuracy of 94.43% for non-COVID-19 viral pneumonia vs. normal and 91.43% for bacterial pneumonia vs. normal and 94.00% accuracy for three-class classification and four-class classification with 93.42% accuracy. The heterogeneity of image sources across public datasets introduced domain shifts affecting consistency and performance in real-world clinical environments.

Avola et al. [17] evaluated twelve DNN architectures using transfer learning for the pneumonia classification from lung radiographic images. Employing two datasets with chest radiographs of patients diagnosed with pneumonia caused by bacterial, generic viral or SARS-CoV-2 infections and unaffected cases. The architecture was adapted for transfer learning by replacing the final classification layers of each model with task-specific dense layers and leveraged hierarchical feature extraction to identify localized pathological patterns. On evaluation in the combined dataset, MobileNetV2 and MobileNetV3 achieved the highest accuracies of above 0.80, followed by AlexNet (0.78), DenseNet and ResNet variants (0.75–0.79) and lower accuracies for SqueezeNet (0.33) and NASNet (0.55). Significant performance degradation was observed for most models under data-scarce conditions.

Zhu et al. [18] suggested a multi-task DL approach for classification of pneumonia types and segmenting associated lesions from chest CT scans involving 181 patients, categorized as lobar, lobular or interstitial pneumonia, while ground truth labels for segmentation were created through manual annotation. The network handled segmentation and classification tasks, streamlining the diagnostic workflow by using shared feature extraction pathways. The multitask architecture enabled the model to contextualize lesion morphology in parallel with disease categorization and achieved a classification accuracy of 92.7%. Bias was introduced by using data from a single medical center that hampered generalizability across diverse populations, imaging protocols and scanner types.

An et al. [19] proposed a deep CNN for pneumonia classification integrating EfficientNetB0 and DenseNet121 as feature extractors. The architecture incorporated multi-head self-attention mechanisms to enhance feature representation and discrimination across varying spatial regions of chest X-ray images. A cross-channel attention-based feature fusion strategy is implemented to combine features from both networks and employed residual blocks and dynamic pooling layers for refining the learning. The attention-enhanced architecture allowed for selective amplification of disease-relevant features, improving the interpretability and clinical reliability of the predictions. The study utilized a chest X-ray dataset that included both normal and pneumonia-affected cases and achieved an accuracy of 95.19%. The exclusive use of static two-dimensional images restricted diagnostic depth compared to three-dimensional volumetric data.

Bhandari et al. [20] presented a DL framework for classifying chest radiograph images into four distinct categories, with pneumonia as one category. The study utilized a single CNN architecture and utilized a publicly available dataset of 7,132 images. The model was integrated with an explainable AI (XAI) framework using SHAP, LIME and Grad-CAM, which offered visual and statistical information on the decision-making method of the classifier. These XAI techniques provided class-discriminative feature maps and model interpretability, which were further corroborated by medical experts, and the interpretive capability added trustworthiness to the diagnostic tool. The system achieved an accuracy of 94.31% through ten-

fold cross-validation, but the dataset size limitations hampered the generalization of the study.

Xue et al. [21] evaluated DL techniques for the segregation of COVID-19 and pneumonia utilizing both thoracic radiograph and computed tomography scan images. The research employed an ensemble of deep transfer learning models: ResNet152, ResNet50, DenseNet121 and an enhanced version of VGG16 across five different datasets. Two datasets contained chest radiographs, while the remaining three included CT imaging data and achieved precision and accuracy ranging from 95% to 96%. The integration of deep residual and densely connected layers allowed for hierarchical feature extraction across both modalities, and transfer learning addressed data scarcity and computational efficiency. Despite its better performance, the study acknowledged a critical limitation related to inconsistencies in ground truth annotations within the datasets, affecting the reliability and generalizability of learned features across different institutions and clinical settings.

Barakat et al. [22] suggested an ML-based framework designed for the early detection of pediatric pneumonia using thoracic radiograph images. The model segmented the input chest X-ray into 64 fixed regions of interest (ROIs) for focused and granular feature extraction. The extracted features were mapped back to their anatomical locations, supporting interpretable diagnostics. On evaluation employing various ML models, the quadratic SVM classifier, using a 64-region-of-interest (ROI) feature extraction scheme, achieved the highest performance with an accuracy of 97.58%, with a lesser classification time compared to transfer learning benchmarks. Extensive reliance on hand-engineered feature extraction and manually defined ROIs that did not generalize across varying image resolutions, anatomical variations and datasets introduced the risk of omitting subtle pathological indicators outside the preselected ROIs and affected model adaptability in practical clinical environments.

A. Research Gap

Despite substantial research in pneumonia detection using AI such as EfficientNetV2L, DenseNet, ResNet50, ViT etc., certain challenges still exist [8] [12] [21]. Most frameworks focus on overall accuracy improvement but fall short to address segmentation of pathological regions, model explainability and clinical adaptability properly [16] [21]. Even though MDEV and ensemble CNNs attempt to enhance feature representation, computational overhead hinders their real-time deployment [9] [14]. Attention-based architectures and multi-head fusion exhibits promising performance but is burdened by the model complexity, making training and interpretability more challenging. Furthermore, segmentation is often excluded, limiting end-to-end diagnostic support and limited external validation also reduces generalizability across diverse clinical environments. A cohesive model that balances both local and global feature extraction capabilities is absent in the majority of the research. Only a few number of existing methods have investigated designs that can carry out simultaneous classification and segmentation with great computing efficiency and diagnostic confidence. Hence, there is a pressing need for a unified, lightweight and interpretable hybrid framework that can simultaneously perform accurate classification and segmentation while being computationally feasible for practical medical applications.

III. Materials and Methods

The proposed model integrates ResNet50 and Swin transformer effectively for the classification and segmentation of thoracic radiographs for Pneumonia. The Chest X-Ray images (Pneumonia) dataset is utilized in the study, and the methodological pipeline consists of a ResNet50 backbone and Swin Transformer modules for hybrid feature extraction after meticulous preprocessing and data augmentation. A segmentation decoder for lesion localization is employed for the segmentation task, and the classification head utilizes global average pooling (GAP) and a multilayer perceptron head (MLP). The proposed model's block diagram is illustrated in Fig. 1.

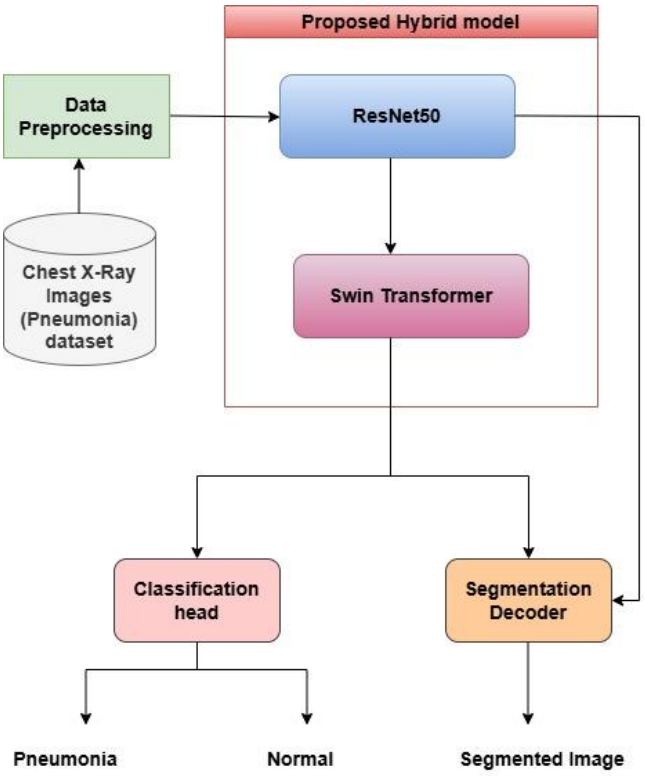


Fig. 1. Block diagram.

B. Dataset Description

The study utilizes Chest X-Ray images (Pneumonia), the openly accessible Kaggle dataset with images gathered from Guangzhou Women and Children's Medical Centre in China [23]. It has 5,863 anterior-posterior (AP) view chest radiographs of paediatric patients aged between one and five years categorized either as Normal or Pneumonia. Fig. 2 represents the number of images present in the three primary directories: train, test and validation, each with two subfolders corresponding to the diagnostic categories. Normal X-rays show clear lungs free of any abnormal opacities. Viral pneumonia primarily presents as more diffuse, bilateral interstitial opacities, while bacterial pneumonia usually exhibits focal lobar consolidation, especially in the upper lobes.

All chest X-rays underwent quality screening to remove unclear or low-resolution scans to guarantee diagnostic accuracy and image credibility. Two expert radiologists independently confirmed the image diagnose and the evaluation was examined by third expert to identify potential discrepancies in classification. This pediatric dataset presents real-world complexity due to the subtle and variable nature of radiographic pneumonia manifestations in young children, making it an ideal benchmark for evaluating AI-based diagnostic systems. Its expert-verified labels, well-structured format and diversity in infection types make it well-suited for both training and evaluation of models focused on classification and lesion localization. Furthermore, the public accessibility facilitates reproducibility, benchmarking and practical deployment of AI solutions in resource-limited or high-volume clinical settings. Fig. 3 illustrates the random sample images in the dataset.

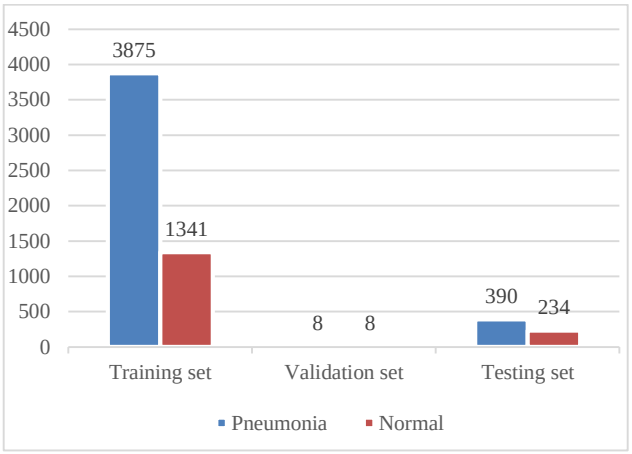


Fig. 2. Number of images in each directory.

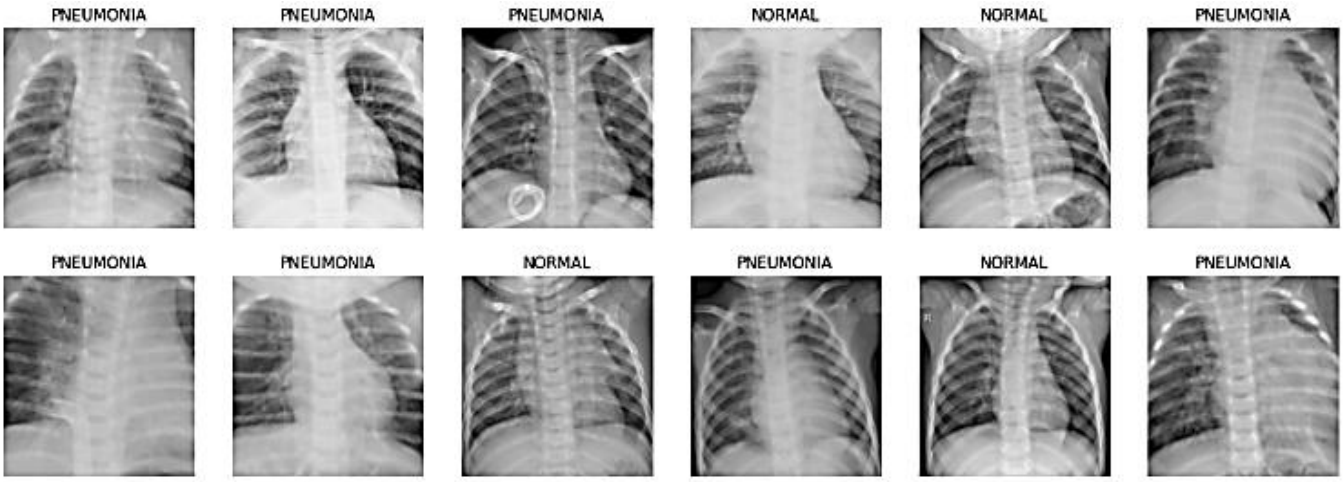


Fig. 3. Random sample images in the dataset.

C. Data Preprocessing

A systematic preprocessing and augmentation strategy is designed for the chest X-ray images prior to inputting them into the model. Initially, each image is resized to a standardized spatial dimension of 224×224 pixels to maintain compatibility with both ResNet50 and Swin Transformer input requirements. The original greyscale chest X-ray image is then converted to a 3-channel RGB image through channel-wise expansion. This transformation is necessary to match the pretrained convolutional backbones that anticipate three-channel inputs. Subsequently, the pixel intensity values $I \in [0,255]$ are normalized to a continuous range of $[0,1]$ using the rescaling operation, as shown in Eq. (1):

$$I_{normalized} = \frac{I}{255} \quad (1)$$

Data augmentation is introduced in the training set to increase the generalization capability and to reduce overfitting. The augmentation operations include horizontal and vertical flipping, random zooming with a zoom range up to 0.3, shearing with a shear angle of 10 degrees and rotation within ±20 degrees. By scaling the intensity in a range of [0.5,2.0], intensity variations are also applied, along with spatial translations. As certain transformations resulted in vacant pixel areas, the nearest neighbour approach is utilized to fill in missing values, as shown in Eq. (2):

$$I'(x,y) = \arg \min_{(x',y')} \|(x,y)(x',y')\| \; ; \; \text{such that I}'(x',y') \neq 0 \quad (2)$$

The images are then organized into mini-batches and passed into the model pipeline. Labels are encoded into binary numerical form, as in Eq. (3):

$$y = \begin{cases} 0 \; ; Normal \\ 1 \; ; Pneumonia \end{cases} \quad (3)$$

In addition to providing enough variation to the training set for reliable feature extraction, the preprocessing pipeline assures the spatial integrity and semantic information preservation of the thoracic radiograph images.

D. Model Development

1) ResNet50: ResNet50 is a fifty-layer deep CNN architecture built upon the principle of residual learning, which resolves the vanishing gradient problem by employing shortcut connections [24]. A stem layer is followed by four sequential convolutional stages, each containing a stack of convolutional blocks and identity blocks. Each block includes a sequence of

convolutional layers with kernel sizes set to 1×1, 3×3 and 1×1, followed by batch normalization and ReLU activation. The defining feature of ResNet50 is the residual connection, where the input x to a block is added to its output F(x), forming the final output, as in Eq. (4):

$$y = F(x, \{W_i\}) + x \quad (4)$$

where, $\{W_i\}$ represents a set of learnable parameters of the convolutional layers. This residual mapping promotes deeper network construction while maintaining productivity and enables the network to learn identity functions. Rich local characteristics can be extracted from images using the ResNet50 backbone, which progressively minimizes spatial dimensions while deepening feature representations. In the proposed hybrid model, only the convolutional and residual blocks of ResNet50 are utilized. Fig. 4 illustrates the general architecture of ResNet50.

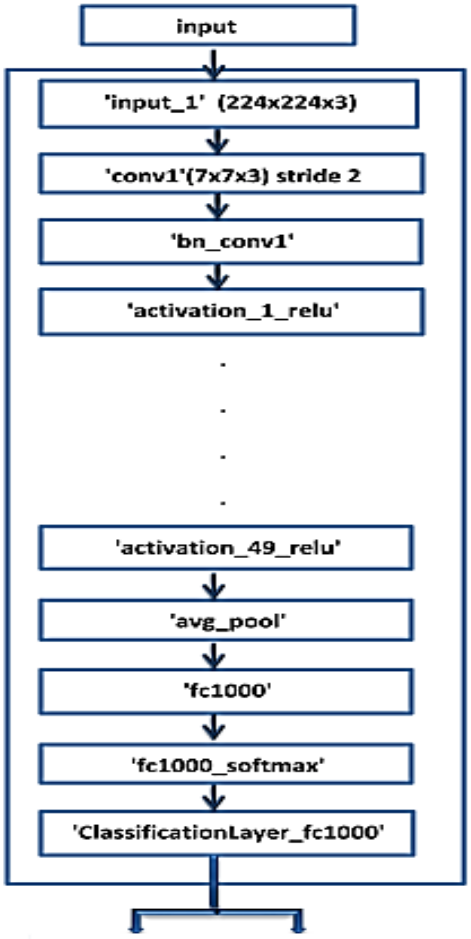


Fig. 4. General architecture of ResNet50.

An input image is initially passed through a convolutional layer with a 7×7 kernel and a stride of 2 to reduce spatial dimensions while capturing broad features. This is followed by batch normalization and a ReLU activation to standardize and introduce non-linearity. The network then progresses through a sequence of 49 residual layers, each built using bottleneck blocks that incorporate skip connections. These connections help preserve input information and facilitate efficient gradient flow, enabling deep network training without degradation. Each residual block performs convolutional transformations and combines the original input with the transformed output. Following the final activation layer, the architecture applies GAP to reduce the spatial dimensions, resulting in a 1×1×2048 feature vector. The vector is then carried to a fully connected dense layer with 1000 units and finally through a softmax activation, producing class probabilities. In the proposed model, the softmax and classification layers are discarded as ResNet50 is used solely for deep feature extraction.

2) Swin transformer: The Swin Transformer (Shifted Window Transformer) is a hierarchical ViT that applies self-attention within non-overlapping local windows and shifts these windows across layers to enable cross-window connections [25]. It is composed of a series of Swin Transformer Blocks, each including Layer Normalization (LN), Multi-head Self-Attention (MSA) within shifted local windows, MLP layers with GELU activation and Residual connections. Fig. 5 represents the basic architecture of the Swin transformer.

Initially, the input image $I \in \mathbb{R}^{H \times W \times 3}$ is partitioned into patches and embedded into a lower-dimensional space, as shown in Eq. (5):

$$x_0 = PatchEmbed(I), x_0 \in \mathbb{R}^{\frac{H}{P} \times \frac{W}{P} \times C} \quad (5)$$

where, P is the patch size and C is the embedding dimension. The embeddings pass through a series of stages composed of Swin Transformer Blocks. The spatial resolution is minimized and the feature dimension is boosted in each stage, resulting in a hierarchical representation. Window-based MSA (W-MSA) followed by Shifted Window MSA (SW-MSA) is applied in each block further. The attention is computed within each window w , as in Eq. (6):

$$Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d}} + B\right)V \quad (6)$$

where Q , K and V represent the query, key and value matrices respectively, computed from the patch tokens confined within a local window and d, the dimension of the key vectors and B, the relative position bias. Each attention output is passed through a LN layer to stabilize the training dynamics, as shown in Eq. (7):

$$LN(x) = \frac{x - \mu}{\sqrt{\sigma^2 + \epsilon}} \cdot \gamma + \beta \quad (7)$$

where, μ, σ represent the mean and standard deviation of the input feature vector respectively and γ, β represents learnable affine parameters. This is followed by a Feed-Forward Network (FFN) comprising two linear transformations with a GELU non-linearity in between, as in Eq. (8):

$$FFN(x) = W_2 \cdot GELU(W_1 \cdot x + b_1) + b_2 \quad (8)$$

where, W_1, W_2 represents weight matrices, b_1, b_2 are biases.

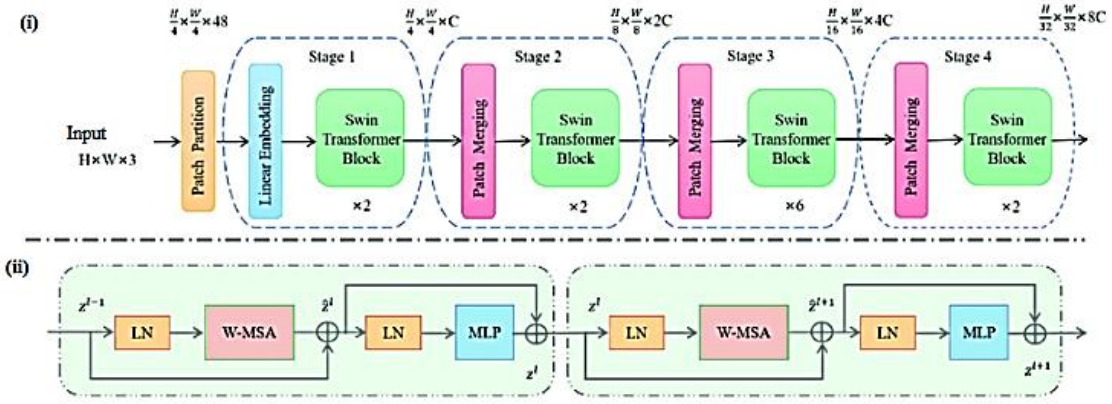


Fig. 5. Swin transformer: (i) Basic architecture (ii) Swin transformer block.

3) Proposed ResNet50- Swin Transformer hybrid model: The proposed hybrid architecture integrates the advantages of both convolutional and transformer-based approaches by employing ResNet50 for localized feature extraction and the Swin Transformer for global context modelling. ResNet50 captures low and mid-level spatial features from the input thoracic radiograph images, identifying pathological patterns in the process. The output feature map of ResNet50 undergoes processing to be compatible with the input requirement of the Swin transformer. A convolutional adaptation layer consisting of a 1×1 followed by batch normalization and ReLU activation is utilized to transform the feature dimensions to the patch embedding size of the Swin Transformer, as shown in Eq. (9):

$$F_{adapted} = RELU\big(BN(W_c * F_{ResNet} + b)\big) \quad (9)$$

where, $W_c \in \mathbb{R}^{1\times1\times c_1\times c}$ represents the kernel for dimension adjustment. c_1, c denotes the number of input and output channels, respectively, F_{ResNet} represents the feature map output from ResNet50, of shape $[H, W, c_1]$, b denotes the learnable bias added after convolution and $*$ represents the convolution operation. Once adapted, the feature maps are tokenized into non-overlapping patches and embedded, forming the input to the Swin Transformer. The window-based and shifted window-based self-attention mechanisms of the transformer architecture allow it to gather long-range dependencies and semantic patterns across the image. Both local features from ResNet50 and global contextual cues from the transformer are therefore jointly modelled. The output of the final Swin Transformer block is bifurcated: one branch passes through a GAP layer followed by an MLP for binary classification (pneumonia or normal), while the other branch serves as the input to a segmentation decoder. The decoder reconstructs high-resolution spatial masks of infected lung regions through a cascade of upsampling and convolutional layers.

a) Classification head: Global Average Pooling (GAP) is a downsampling operation that reduces each feature map to a single scalar by computing the average of all spatial elements converting feature maps into compact feature vectors while preserving spatially aggregated information [26]. In the proposed model, the GAP layer condenses the spatial dimensions of the feature map output of the Swin Transformer into a single feature vector. This operation preserves the most salient global contextual features and drastically minimises the number of parameters and computation before classification. The resultant feature map of dimension $H \times W \times C$ of the Swin transformer block is spatially reduced to produce a compact descriptor vector, as in Eq. (10):

$$x_{GAP} = \frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W} F_{Swin}[i,j] \quad (10)$$

The condensed descriptor is then passed into a final MLP layer that serves as the final decision-making unit by transforming the aggregated features into a scalar output that facilitates binary classification. Its learnable weights and bias terms are optimized during training to minimize loss and improve diagnostic accuracy. A sigmoid function is applied to generate a probability score $\hat{y} \in [0,1]$ activated output is as shown in Eq. (11):

$$\hat{y} = \sigma(W \cdot x_{GAP} + b) \quad (11)$$

where, W and b denote the weight and bias of the dense layer, and the sigmoid function is denoted by σ.

b) Segmented decoder: The segmentation decoder is responsible for reconstructing pixel-wise masks from high-dimensional feature representations learned by the encoder, a combined sequence of ResNet50 and Swin Transformer. During the forward pass, selected intermediate feature maps from ResNet50 are cached and routed via skip connections to the corresponding stages in the decoder. This facilitates the recovery of fine spatial details lost during downsampling and enhances localization precision. The decoder adopts an up-sampling architecture that consists of bilinear upsampling followed by convolutional blocks that refine spatial granularity. The up-sampled feature at level l is obtained, as shown in Eq. (12):

$$F_{up}^{(l)} = Upsample\left(F_{dec}^{(l+1)}\right) \quad (12)$$

where, $F_{dec}^{(l+1)}$ is the decoder feature map from a deeper layer $l+1$. The output is concatenated with the corresponding encoder feature map $F_{enc}^{(l)}$ from ResNet50, as shown in Eq. (13):

$$F_{concat}^{(l)} = Concat\left(F_{up}^{(l)}, F_{enc}^{(l)}\right) \quad (13)$$

The concatenated feature is processed through a convolutional refinement block, as in Eq. (14):

$$F_{dec}^{(l)} = ReLU\left(BN\left(W^{(l)} * F_{concat}^{(l)} + b^{(l)}\right)\right) \quad (14)$$

where, $W^{(l)}$ and $b^{(l)}$ are the convolution weights and biases, BN represents batch normalization, and $ReLU$ is the non-linear activation function. The decoding process is iteratively performed through successive layers until the output spatial resolution is comparable to that of the input image. Finally, a 1×1 convolutional layer is applied to map the refined feature maps to a single-channel output, as shown in Eq. (15):

$$S = \sigma(W_{seg} * F_{dec}^{(l)} + b_{seg}) \quad (15)$$

where, W_{seg} and b_{seg} denote weights and bias of the final layer. The segmentation output $S \in [0,1]^{H\times W}$ indicates the prospect of pneumonia-affected regions for each pixel in the input image. The architecture enables effective integration of global context via Swin Transformer and spatial precision via ResNet skip features, resulting in reliable and interpretable pneumonia lesion segmentation. Fig. 6 illustrates the basic architecture of the proposed ResNet50-Swin Transformer hybrid model. The detailed algorithm for the proposed model is given below (see Algorithm 1).

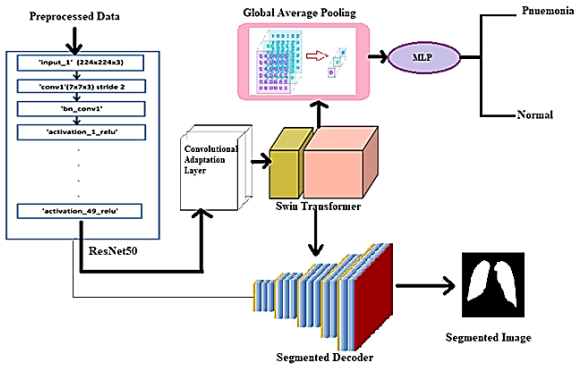


Fig. 6. Basic architecture of the proposed ResNet50-swin transformer hybrid model.

Algorithm 1: ResNet50-Swin Transformer hybrid model

Input:

Chest X-ray image I ∈ $\mathbb{R}^{H\times W\times 3}$

Label vector $y_i \in [0,1]$, indicating Pneumonia (1) or Normal (0)

Output:

Predicted class label $\hat{y} \in [0,1]$

Segmentation mask $S \in \mathbb{R}^{H\times W}$

Begin:

Data collection

Load Chest X-ray image (Pneumonia) dataset

Pre-processing

Resize the images into 224x224

Normalize pixel values to range [0,1]: $I_{normalized} = \frac{I}{255}$

Label encoding in the images:

$$y = \begin{cases} 0 \ ; Normal \\ 1 \ ; Pneumonia \end{cases}$$

Data Augmentation on the training dataset

ResNet50

Local feature extraction and final output generation, $y = F(x,\{W_i\}) + x$

Store feature maps for decoder skip connections.

Feature Adaptation

Transform F_{ResNet} to match Swin Transformer input size:

$F_{adapted} = RELU\big(BN(W_c * F_{ResNet} + b)\big)$

Swin Transformer

Embed $F_{adapted}$ into non-overlapping patches.

Apply Swin Transformer blocks with shifted window attention and FFN:

$Attention(Q,K,V) = Softmax\left(\frac{QK^T}{\sqrt{d}} + B\right)V$

Apply Layer Norm → FFN → Layer Norm sequence

$LN(x) = \frac{x-\mu}{\sqrt{\sigma^2+\epsilon}} \cdot \gamma + \beta \ ; \ FFN(x) = W_2 \cdot GELU(W_1 \cdot x + b_1) + b_2$

Store final feature map F_{Swin}

Classification Head

Apply Global Average Pooling:

$x_{GAP} = \frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W} F_{Swin}[i,j]$

Multilayer Perceptron apply sigmoid function:

$\hat{y} = \sigma(W \cdot x_{GAP} + b)$

Segmentation Decoder

Input: F_{Swin} and skip connections from ResNet50.

For each decoder stage:

Up sample previous output

Concatenate with corresponding skip connection

Apply convolution → BatchNorm → ReLU

Final segmentation masks=σ (Conv1×1 (Decoder Output))

Model Compilation and Training

Compile model with loss = Binary crossentropy, learning rate = 0.0001, optimizer = Adam, Epochs =75

Train model: model.fit (X_train, y_train)

Evaluation and Model Saving

Evaluate model: model. evaluate (X_test, y_test)

Tune hyperparameters

Save the model

End

E. Simulation Map

The proposed hybrid model was implemented in a high-level deep learning framework and trained on a workstation equipped with a high-performance GPU (NVIDIA RTX 3090, 24GB VRAM), 128GB RAM and an Intel Xeon processor. Python was used as the programming language, and model development was carried out using the Keras API built on top of TensorFlow, chosen for its modular design, scalability and ease of customization for hybrid DL architectures. Training and experimentation were conducted on the Google Colaboratory (Colab) platform, leveraging its cloud-based infrastructure and access to high-performance GPUs. The training process was optimized by carefully tuning key hyperparameters, which significantly influenced model convergence and generalization. Table I shows the full list of hyperparameters and training settings employed in the study.

TABLE I. HYPERPARAMETER SPECIFICATIONS

Hyper parameters	Values
Optimizer	ADAM
Swin Transformer Layers	4
Loss function	Binary cross-entropy (Classification) Dice loss (Segmentation)
Activation function	Sigmoid
Batch size	16
Epochs	75
Learning Rate	0.0001
Dropout	0.3

IV. RESULTS

A set of standard evaluation metrics, as illustrated in Eq. (16) to Eq. (20), was employed to comprehensively assess the proposed model's performance. True Positives (TP), False Positives (FP), True Negatives (TN) and False Negatives (FN) are mathematically computed using the confusion matrix core elements. Different metrics offer unique insights into the model performance. Overall correctness by accuracy, precision and recall highlights the ability of the model to correctly detect pneumonia without excessive misses or false alarms.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (16)$$

$$Precision = \frac{TP}{TP+FP} \quad (17)$$

$$Recall = \frac{TP}{TP+FN} \quad (18)$$

$$F1-score = 2 \times \frac{precision \times Recall}{Precision+Recall} \quad (19)$$

The Dice Coefficient evaluates the similarity between predicted segmentation and the ground truth and measures the degree of overlap between two binary masks and ranges from 0 to 1.

$$Dice(A,B) = \frac{2|A \cap B|}{|A|+|B|} \quad (20)$$

where, A and B denote the sets of predicted and ground truth pixels, respectively, labelled as positive. A dice coefficient close to 1 indicates better spatial agreement between prediction and ground truth. The accuracy and loss plots serve as fundamental diagnostic tools to monitor the model's learning dynamics over successive training epochs. A steadily increasing accuracy curve alongside a consistently decreasing loss trajectory indicates effective gradient optimization and convergence toward a generalized solution. Any divergence between training and validation curves may suggest overfitting or underfitting, necessitating hyperparameter adjustments.

The accuracy plot of the proposed hybrid model, represented in Fig. 7, demonstrates a strong and consistent learning curve over 75 epochs, highlighting effective training convergence. A rapid rise in training and validation accuracies is observed within the initial 20 epochs, indicative of efficient low-level feature acquisition. As the epochs progress, the model's accuracy stabilizes with the training accuracy approaching 99.5%, while the validation accuracy maintains close proximity, around 98.4%. The minimal gap between the two curves signifies robust generalization and minimal overfitting and affirms the model's potential in retaining discriminative features and achieving high classification reliability on unseen chest X-ray data.

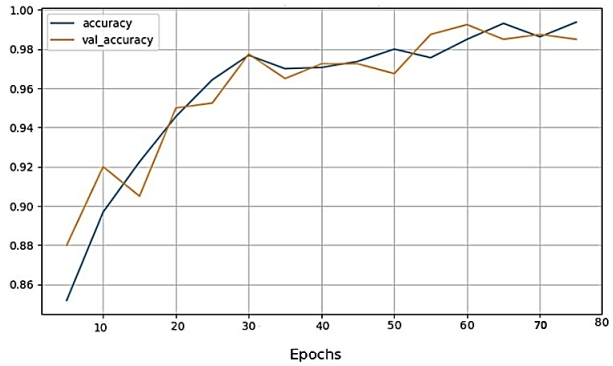


Fig. 7. Accuracy plot of ResNet50-Swin Transformer hybrid model.

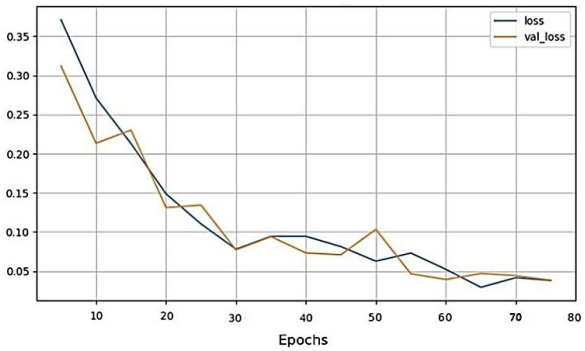


Fig. 8. Loss plot of ResNet50-Swin Transformer hybrid model.

Fig. 8 illustrates the loss plot for the proposed model that demonstrates a steady and significant decrease in both training and validation loss over 75 epochs, reflecting effective model convergence. In the initial training phase, a steep decline in loss is observed during the first 15 epochs, indicating rapid optimization of weights and improved learning of discriminative features. Beyond epoch 20, the loss values continue to decrease gradually and stabilize around 0.04, signifying that the model is

minimizing the error function efficiently. The close alignment between training and validation loss trajectories throughout the epochs confirms the absence of overfitting and affirms the generalization capability across unseen data.

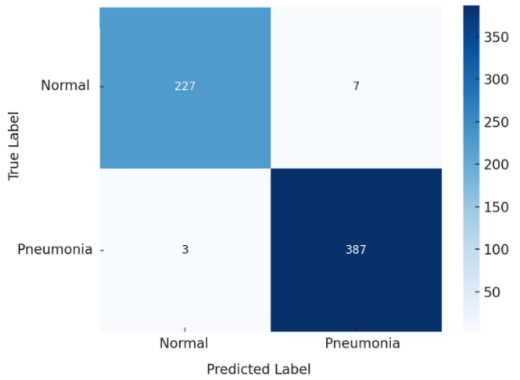


Fig. 9. Confusion matrix of the proposed model.

Fig. 9 represents the confusion matrix highlighting the classification performance of the hybrid model on the test dataset. Out of 234 normal images, the framework achieved a true negative rate of around 97%, and in the pneumonia category, 387 out of 390 pneumonia images were correctly classified, exhibiting better accuracy. The results indicate strong predictive performance and class discrimination, particularly in identifying pneumonia cases, and the minimal number of misclassifications further emphasizes the model's reliability for binary medical image segregation.

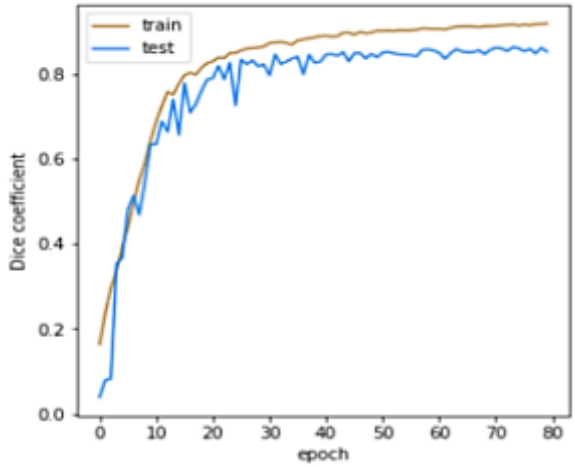


Fig. 10. Dice coefficient plot.

The Dice Coefficient plot illustrated in Fig. 10 demonstrates the segmentation performance of the proposed model over 75 training epochs. Both training and test curves exhibit a steep ascent within the initial 20 epochs, indicating rapid learning and effective optimization of segmentation boundaries. Beyond epoch 25, the Dice Coefficient stabilizes around 0.88 for the test set, while the training score plateaus slightly higher, suggesting excellent overlap between predicted and ground truth segmentation masks with minimal overfitting. The narrow gap between training and test curves throughout the learning process reflects the model's excellent consistency across datasets. The stable performance at later epochs confirms that the model effectively captures spatial and structural patterns necessary for precise medical image segmentation, reinforcing its reliability in real-world diagnostic contexts.

	precision	recall	f1-score
Pneumonia	0.98	0.98	0.97
Normal	0.98	0.99	0.98
accuracy			0.98
macro avg	0.98	0.99	0.98
weighted avg	0.98	0.99	0.98

Fig. 11. Classification report of ResNet50-Swin Transformer hybrid model.

Fig. 11 represents the classification report for the proposed framework, revealing outstanding predictive performance across both classes. With a precision of 0.98 for both classes, the model has a high proportion of accurately detected positive instances relative to all predicted positives. The recall for the Normal class reaches 0.99, while Pneumonia records a recall of 0.98, demonstrating the excellent sensitivity in detecting true positives, particularly for pneumonia diagnosis. The F1-score stands at 0.97 for Pneumonia and 0.98 for Normal, highlighting the model's balanced and reliable classification performance, and the overall accuracy of 0.984 reaffirms the model's robustness. These results collectively validate the clinical reliability of the proposed model in distinguishing pneumonia from normal thoracic radiograph images with great precision and generalizability. Fig. 12 represents the evaluation metrics of the proposed model.

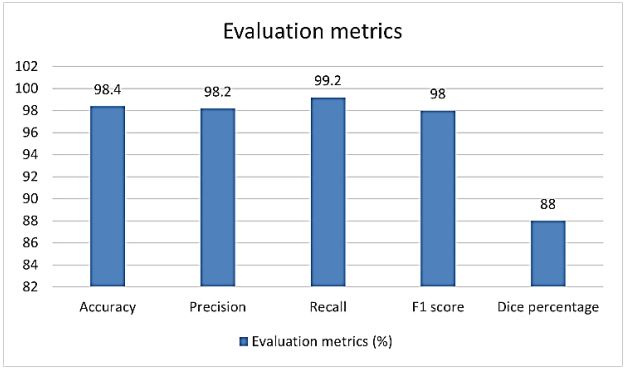


Fig. 12. Evaluation metrics for the proposed model.

The visual outputs generated by the proposed hybrid model are illustrated in Fig. 13 and Fig. 14, effectively demonstrating its dual capabilities in both classification and segmentation of chest X-ray images. As evident from the classification report, the framework accurately distinguishes between pneumonia and normal cases, with predictions aligning closely with the actual ground truth labels, while the segmentation results highlight the effectiveness in accurately localizing lung regions. The original mask and the segmented output exhibit strong overlap, with the segmented regions closely approximating the annotated ground truth. While some minor deviations and false-positive activations are observed at the lung peripheries, the overall shape, structure and spatial consistency of the lung regions are

well-preserved. These outcomes confirm the robustness of the segmentation decoder in delineating pulmonary areas, thereby supporting further clinical interpretability and diagnostic localization.

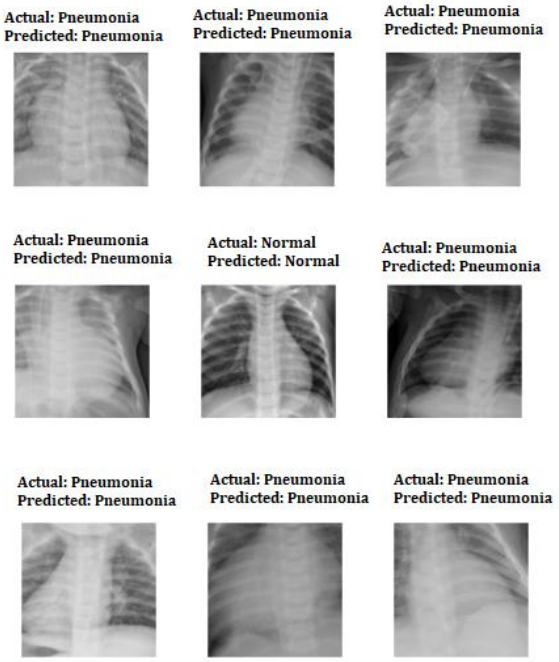


Fig. 13. Classification results.

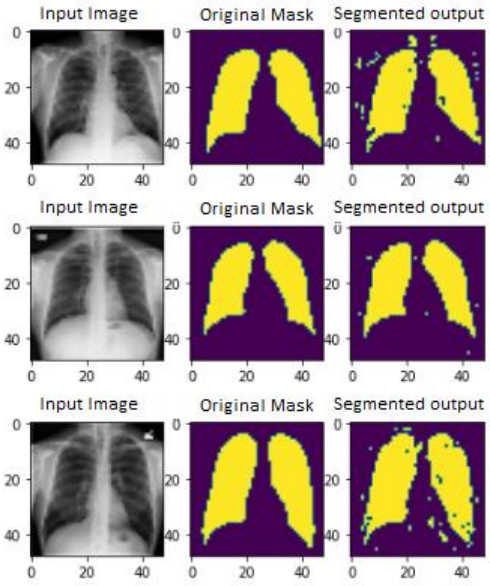


Fig. 14. Segmented output.

TABLE II. ACCURACY COMPARISON OF RESNET50-SWIN TRANSFORMER HYBRID MODEL WITH OTHER METHODS

Author [Ref]	Model	Accuracy
Ali et al. [8]	EfficientNetV2L	94.02%
Shaikh et al. [9]	MDEV	92.15%
Wang et al. [11]	DenseNet with SE blocks	92.8%
Mabrouk et al. [12]	Ensemble Learning	93.91%
Ortiz-Toro et al. [13]	KNN, SVM, RF	91.3%
Bhatt et al. [14]	Ensemble CNNs	84.12%
Singh et al. [15]	Attention-aware CNN	95.47%
Ibrahim et al. [16]	AlexNet	94.43%
Avola et al. [17]	MobileNetV3	80%
Zhu et al. [18]	Multi-task DL	92.7%
An et al. [19]	EfficientNetB0 + DenseNet121	95.19%
Bhandari et al. [20]	CNN + XAI	94.31%
Xue et al. [21]	Ensemble	96%
Barakat et al. [22]	Quadratic SVM	97.58%
Proposed model	**ResNet50-Swin Transformer hybrid**	**98.4%**

Table II and Fig. 15 illustrate the comparative analysis of the proposed hybrid framework with conventional pneumonia detection models, revealing that even though some approaches demonstrate comparable classification accuracy, they often suffer from drawbacks in terms of interpretability, architectural complexity and generalizability. MDEV and ensemble CNN architecture models were efficient in capturing diverse features, but are hampered by computational overhead due to their multi-network designs restricting real-time installation, especially in limited resource settings. Radiomics and SVM-based frameworks, relying on hand-crafted features or region-of-interest segmentation struggles with adaptability across varying anatomical structures and image qualities and were seen as inefficient in most clinical scenarios. The better performing models such as EfficientNetB0 + DenseNet121 or attention-aware CNNs, with accuracies in the range of 94 to 96%, require complex attention modules or fine-tuned fusion strategies increasing training time and computational complexity limiting their adaptability. The proposed ResNet50-Swin Transformer hybrid model achieves superior accuracy of 98.4%, combining the convolutional strength of ResNet50 for low-level feature extraction with the hierarchical attention-driven capabilities of Swin Transformer for contextual learning. The framework not only enhances classification precision but also enables segmentation functionality within a unified architecture, delivering both diagnostic accuracy and spatial interpretability in a computationally efficient manner.

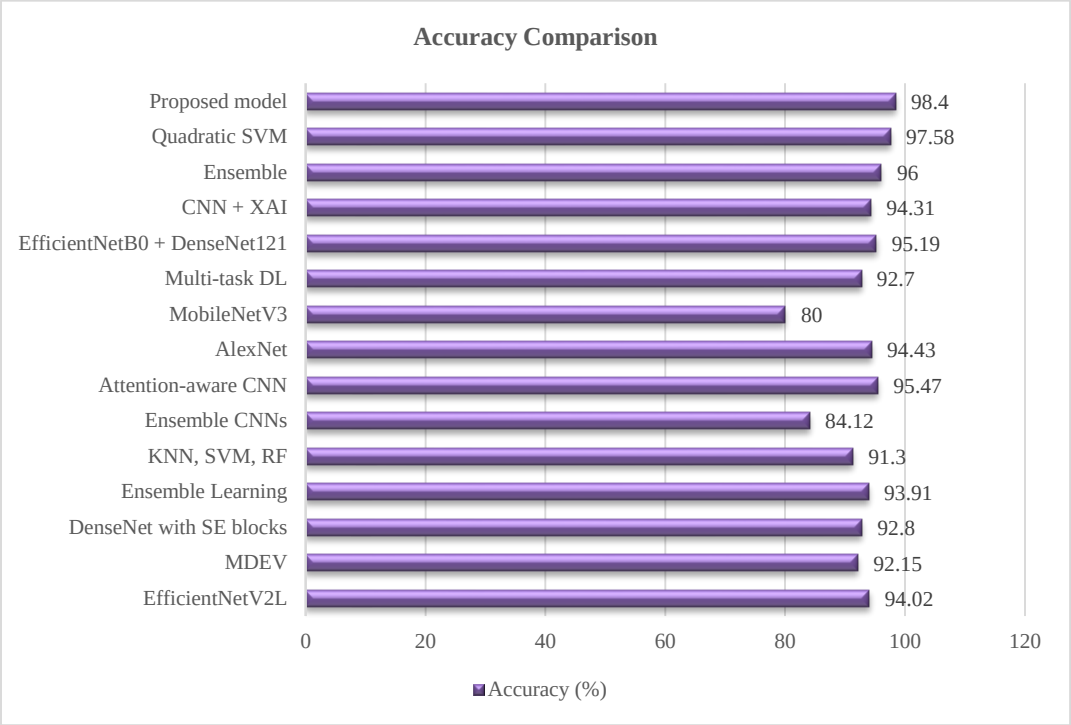


Fig. 15. Accuracy comparison of ResNet50-Swin Transformer hybrid model with other methods.

V. Discussions

The hybrid ResNet50-Swin Transformer framework demonstrates a novel architecture for dual-task learning, enabling both binary classification and pixel-level lesion localization of pneumonia from thoracic radiographs. Unlike conventional models that excel in feature extraction but often lack contextual awareness, the incorporation of the Swin Transformer effectively models global dependencies through its hierarchical window-based attention mechanism. This context-aware learning facilitates improved representation of complex pulmonary structures, especially in cases with overlapping anatomical patterns or ambiguous radiographic findings. The framework's superior classification accuracy (98.4%) and high Dice Coefficient (0.88) for lesion segmentation reinforce the robustness in both predictive performance and clinical interpretability. Compared to existing models that often perform one task effectively but compromise the other, the dual-capability approach ensures diagnostic relevance without incurring excessive computational burden. This efficiency renders it particularly suitable for scalable deployment in clinical settings where radiologist availability or diagnostic infrastructure is constrained. Moreover, the segmentation decoder not only assists in validating classification outputs but also promotes explainability; bridging the trust gap between black-box AI models and medical practitioners. These visual insights are vital for treatment planning and follow-up monitoring, as the hybrid framework offers a pragmatic solution for intelligent diagnostic assistance.

VI. Conclusion

The study presents a novel hybrid DL model integrating ResNet50 with Swin Transformer to simultaneously perform classification and segmentation of pneumonia from thoracic radiograph images. The local feature extraction capability of ResNet50 and the hierarchical attention-driven global context modelling of the Swin Transformer are utilized to skillfully gather both fine-grained and contextual features. The model further branches into a classification head comprising global average pooling and a multilayer perceptron and a segmentation decoder that reconstructs pixel-level patches, providing interpretable diagnostic outputs. With 98.4% accuracy, a precision of 98.2%, a recall of 99.2% and an F1-score of 98% in classification and a Dice coefficient of 0.88 for segmentation, the model outperformed the conventional methods by broad margins.

The segmentation results illustrated accurate delineation of lung regions, aligning closely with the ground truth masks, indicating exceptional interpretability and potential applicability in aiding radiologists. Additionally, the model's dual functionality can streamline radiology workflows by reducing dependency on separate diagnostic and localization tools, thereby saving time and operational costs.

However, the study is not without limitations. The dataset used is limited to pediatric chest radiographs collected from a single institution, which may impact the model's generalizability to broader populations, including adults or those with coexisting pulmonary conditions. Additionally, the segmentation performance may degrade in cases of overlapping pathologies or poor image quality. These limitations call for broader validation to ensure clinical robustness.

Future extensions of this work may focus on multiclass classification involving multiple thoracic diseases, incorporating explainable AI modules to enhance transparency and validating the framework across multi-institutional and

multi-demographic datasets to ensure broader generalizability and adoption. Further, the integration into mobile diagnostic applications or point-of-care systems could significantly expand its reach to rural and underserved areas.

REFERENCES

[1] R. Karim, J. K. Afridi, S. R. Yar, M. B. Zaman, and B. K. Afridi, "Clinical findings and radiological evaluation of WHO-defined severe pneumonia among hospitalized children," Cureus, vol. 15, no. 1, Jan. 2023.C.

[2] Stotts, V. F. Corrales-Medina, and K. J. Rayner, "Pneumonia-induced inflammation, resolution and cardiovascular disease: causes, consequences and clinical opportunities," Circ. Res., vol. 132, no. 6, pp. 751–774, Mar. 2023.

[3] R. K. Seramo, S. M. Awol, Y. A. Wabe, and M. M. Ali, "Determinants of pneumonia among children attending public health facilities in Worabe town," Sci. Rep., vol. 12, no. 1, p. 6175, Apr. 2022.

[4] S. Ahmed, S. Sultana, A. M. Khan, M. S. Islam, G. M. Habib, I. M. McLane, and H. Nair, "Digital auscultation as a diagnostic aid to detect childhood pneumonia: A systematic review," J. Glob. Health, vol. 12, p. 04033, Dec. 2022.

[5] S. Ramgopal, L. Ambroggio, D. Lorenz, S. S. Shah, R. M. Ruddy, and T. A. Florin, "A prediction model for pediatric radiographic pneumonia," Pediatrics, vol. 149, no. 1, Jan. 2022.

[6] J. Zhang, Y. Zhu, Y. Zhou, F. Gao, X. Qiu, J. Li, and W. Lin, "Pediatric adenovirus pneumonia: clinical practice and current treatment," Front. Med., vol. 10, p. 1207568, Jan. 2023.

[7] K. Stokes, R. Castaldo, C. Federici, S. Pagliara, A. Maccaro, F. Cappuccio, and L. Pecchia, "The use of artificial intelligence systems in diagnosis of pneumonia via signs and symptoms: A systematic review," Biomed. Signal Process. Control, vol. 72, p. 103325, Oct. 2022.

[8] M. Ali, M. Shahroz, U. Akram, M. F. Mushtaq, S. C. Altamiranda, S. A. Obregon, and I. Ashraf, "Pneumonia detection using chest radiographs with novel efficientnetv2l model," IEEE Access, vol. 12, pp. xxxx–xxxx, Jan. 2024.

[9] M. Shaikh, I. F. Siddiqui, Q. Arain, J. Koo, M. A. Unar, and N. M. F. Qureshi, "MDEV model: A novel ensemble-based transfer learning approach for pneumonia classification using CXR images," Comput. Syst. Sci. Eng., vol. 46, no. 1, pp. 287–302, Jan. 2023.

[10] A. M. Barhoom and S. S. Abu-Naser, "Diagnosis of pneumonia using deep learning," Int. J. Acad. Eng. Res., vol. 6, no. 7, pp. 12–20, Jul. 2022.

[11] K. Wang, P. Jiang, J. Meng, and X. Jiang, "Attention-based DenseNet for pneumonia classification," IRBM, vol. 43, no. 5, pp. 479–485, Oct. 2022.

[12] A. Mabrouk, R. P. Diaz Redondo, A. Dahou, M. Abd Elaziz, and M. Kayed, "Pneumonia detection on chest X-ray images using ensemble of deep convolutional neural networks," Appl. Sci., vol. 12, no. 13, p. 6448, Jul. 2022.

[13] C. Ortiz-Toro, A. García-Pedrero, M. Lillo-Saavedra, and C. Gonzalo-Martin, "Automatic detection of pneumonia in chest X-ray images using textural features," Comput. Biol. Med., vol. 145, p. 105466, Jun. 2022.

[14] H. Bhatt and M. Shah, "A Convolutional Neural Network ensemble model for pneumonia detection using chest X-ray images," Healthc. Anal., vol. 3, p. 100176, Mar. 2023.

[15] S. Singh, S. S. Rawat, M. Gupta, B. K. Tripathi, F. Alanzi, A. Majumdar, and O. Thinnukool, "Deep attention network for pneumonia detection using chest X-ray images," Comput. Mater. Contin., vol. 74, pp. 1673–1691, May 2023.

[16] A. U. Ibrahim, M. Ozsoz, S. Serte, F. Al-Turjman, and P. S. Yakoi, "Pneumonia classification using deep learning from chest X-ray images during COVID-19," Cogn. Comput., vol. 16, no. 4, pp. 1589–1601, Apr. 2024.

[17] D. Avola, A. Bacciu, L. Cinque, A. Fagioli, M. R. Marini, and R. Taiello, "Study on transfer learning capabilities for pneumonia classification in chest-X-rays images," Comput. Methods Programs Biomed., vol. 221, p. 106833, Nov. 2022.

[18] Q. Zhu, P. Che, M. Li, W. Guo, K. Ye, W. Yin, and S. Li, "Artificial intelligence for segmentation and classification of lobar, lobular, and interstitial pneumonia using case-specific CT information," Quant. Imaging Med. Surg., vol. 14, no. 1, pp. 579–592, Jan. 2023.

[19] Q. An, W. Chen, and W. Shao, "A deep convolutional neural network for pneumonia detection in X-ray images with attention ensemble," Diagnostics, vol. 14, no. 4, p. 390, Feb. 2024.

[20] M. Bhandari, T. B. Shahi, B. Siku, and A. Neupane, "Explanatory classification of CXR images into COVID-19, pneumonia and tuberculosis using deep learning and XAI," Comput. Biol. Med., vol. 150, p. 106156, Nov. 2022.

[21] X. Xue, S. Chinnaperumal, G. M. Abdulsahib, R. R. Manyam, R. Marappan, S. K. Raju, and O. I. Khalaf, "Design and analysis of a deep learning ensemble framework model for the detection of COVID-19 and pneumonia using large-scale CT scan and X-ray image datasets," Bioengineering, vol. 10, no. 3, p. 363, Mar. 2023.

[22] N. Barakat, M. Awad, and B. A. Abu-Nabah, "A machine learning approach on chest X-rays for pediatric pneumonia detection," Digit. Health, vol. 9, p. 20552076231180008, Apr. 2023.

[23] https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia/data.

[24] M. B. Hossain, S. H. S. Iqbal, M. M. Islam, M. N. Akhtar, and I. H. Sarker, "Transfer learning with fine-tuned deep CNN ResNet50 model for classifying COVID-19 from chest X-ray images," Inform. Med. Unlocked, vol. 30, p. 100916, Oct. 2022.

[25] Y. Ma and W. Lv, "Identification of pneumonia in chest X-ray image based on transformer," Int. J. Antennas Propag., vol. 2022, no. 1, p. 5072666, Jan. 2022.

[26] S. Haseli Golzar, H. Bagherpour, and J. Amiri Parian, "A new method to optimize deep CNN model for classification of regular cucumber based on global average pooling," J. Food Process. Preserv., vol. 2024, no. 1, p. 5818803, Jan. 2024.

Developing ReAdaBalancer for Load Balancing Optimization in Networked Cloud Computing

M Diarmansyah Batubara[1]*, Poltak Sihombing[2], Syahril Efendi[3], Suherman[4]
Student of Doctoral Program in Computer Science[1]
Department of Computer Science-Faculty of Computer Science and Information Technology,
Universitas Sumatera Utara, Medan, Indonesia[1, 2, 3]
Department of Electrical Engineering-Faculty of Engineering, Universitas Sumatera Utara, Medan, Indonesia[4]

***Abstract*—Traditional load balancing systems frequently have trouble adjusting to abrupt and unexpected changes in traffic. This can cause problems like server overload, longer response times, and more requests being denied. This problem is highly important in areas like healthcare, finance, cloud computing, and e-commerce, where performance, stability, and fast data delivery are all very important. To solve this problem, this study presents ReAdaBalancer, an adaptive load balancing architecture that aims to improve system performance, scalability, stability, and efficiency in contexts with changing traffic. Flask serves as the backend framework for ReAdaBalancer, while Nginx serves as the load balancer. Real-time monitoring and analytics are used to improve traffic distribution based on the resources that are currently available. Leveraging queuing theory (M/M/s/K Network), the system's performance is tested under diverse load situations, providing insights into its scalability and efficiency. ReAdaBalancer can also learn and adapt all the time, thanks to machine learning and heuristic optimization. This makes sure that it works the same way even when demand changes. Experimental results demonstrate that, under equivalent settings, ReAdaBalancer decreases response times by over 67% and reduces request denial rates by over 50% in comparison to traditional methods. This work has multiple opportunities for subsequent investigation. Future improvements could involve making ReAdaBalancer work in distributed multi-data center environments, adding reinforcement learning to make decisions more independently, looking into load balancing strategies that use less energy, and making it work in edge computing and IoT ecosystems.**

***Keywords*—Cloud computing; heuristic optimization; adaptive load balancing; scalability; ReAdaBalancer**

I. INTRODUCTION

Cloud computing has gained significant popularity in recent years due to the rapid advancement of information and communication technology, which has driven the growth of data centers with resource virtualization and high connectivity [1]. To handle the increasing complexity of cloud environments, this study proposes the Rock Hyrax approach, an algorithmic model inspired by the adaptive and collaborative behavior of hyrax animals. The approach is designed to optimize load balancing in cloud computing systems, ensuring flexibility, scalability, and continuous access to resources [2]. Efficient load management is essential for maintaining performance while reducing resource waste. The Rock Hyrax model mimics the collective intelligence of hyrax groups, allowing computing systems to adapt dynamically to fluctuating workloads through intelligent collaboration. However, the rapid expansion of cloud services also introduces challenges such as system reliability, latency reduction, and energy efficiency. Efficient cloud computing faces several challenges, including latency, dark data risks, governance issues, data availability, privacy, long-term costs, and potential threats from edge computing and related technologies [3]. One promising solution to address these challenges is the implementation of efficient load-balancing techniques. Among innovative approaches, the Rock Hyrax method stands out by emphasizing adaptive, collective strategies inspired by the social behavior of hyrax herds, which are adept at distributing risks and surviving in dynamic environments.

In cloud computing, load balancing distributes workloads across smaller nodes, enhancing system performance and stability [4, 5]. The Rock Hyrax strategy draws inspiration from the cooperative survival tactics of hyrax animals, which allocate roles and adapt effectively in unstable habitats. Existing load balancing algorithms can be broadly categorized into three groups: static, dynamic, and nature-inspired methods [5]. To address modern networked cloud environments, newer approaches must be smarter, more efficient, and adaptive.

Static Load Balancing (SLB) leverages prior knowledge of system resources such as memory, storage, and processing power, resulting in lower overhead compared to dynamic methods [6, 7]. However, SLB lacks flexibility in adapting to changing workloads. Conversely, Dynamic Load Balancing (DLB) offers higher adaptability by adjusting workloads in real time, but at the cost of increased overhead [7]. Nature-inspired approaches, such as the Rock Hyrax algorithm, integrate the strengths of both models by mimicking decentralized, socially adaptive survival strategies. This enables accurate workload distribution with reduced overhead, even under dynamic and uncertain system conditions.

In contrast, Nature Inspired Load Balancing (NLB) algorithms mimic biological and natural processes such as genetic evolution and honey bee foraging [8]. Compared to traditional methods, NLB is generally more effective in solving highly complex problems by approaching global optima rather than local ones. For example, Round Robin, although widely applied in cloud computing, often struggles with scalability and adaptability in dynamic environments. With the growing number of users, scheduling and load balancing techniques have become essential to sustain the performance of cloud

*Corresponding Author.

networks [9]. In this context, the Rock Hyrax approach offers advantages in social adaptability and efficient load distribution, even under unstable network conditions.

Load balancing also plays a key role in optimizing task assignment to virtual machines according to priority [10]. In static implementations, cluster nodes are connected through fixed routing and exchange specialized information for advanced processing [11]. By contrast, dynamic algorithms monitor traffic conditions in real time and redistribute workloads adaptively. This continuous feedback mechanism improves resource utilization, performance, and response time without relying on prior system information [12, 13].

Nature-inspired methods are becoming increasingly popular in this area. For instance, Particle Swarm Optimization (PSO) simulates the collective behavior of fish schools or bird flocks in search of optimal solutions [5, 6, 14]. Other inspirations include the human immune system, which adaptively detects and isolates threats, and physical phenomena such as earthquakes or avalanches, which inform automated post-failure recovery models [15]. Since evolutionary principles can be applied independently of the underlying medium, they enable the development of highly adaptive cloud systems [16].

This study also highlights the theoretical importance of latency and accessibility in cloud load balancing. By modeling more complex dynamic systems, it becomes possible to maximize workload distribution while considering energy efficiency. Therefore, exploring Rock Hyrax and Regret Minimization approaches is highly relevant, as they can optimize resource allocation, reduce energy consumption, and enhance the operational efficiency of machine learning–based applications, including evolving recommendation systems.

This study is organized as follows: Section I introduces the background and related work, Section II describes the design and methodology of ReAdaBalancer, Section III presents the implementation details and experimental setup and also discusses the results and analysis. Finally, Section IV concludes the study with key findings and directions for future research.

II. Materials and Methods

A. Cloud Computing

Cloud computing has changed the way businesses use and manage their IT resources. Now, they focus on ensuring that resources are used efficiently and the load is balanced to get the best performance. Research conducted a thorough review of cloud computing usage and emphasized how important load balancing is for cloud service performance [17]. Dynamic load balancing approaches are becoming increasingly important as cloud setups get more complicated. This approach helps to make resources more efficient and reduce system overhead. Researchers wrote a long report on dynamic load balancing in cloud computing. They talked about how difficult it is for cloud service providers to keep their services available and reliable while also avoiding resource overload [18]. The research suggested using nature-inspired metaheuristic algorithms to improve load balancing [19]. This shows that nature-inspired methods can be used to find flexible solutions in changing contexts. All these studies underline the importance of using advanced optimization algorithms to solve load distribution problems in cloud computing systems.

Despite advancements in cloud computing load balancing, there is still a big gap in getting the best resource distribution in networked systems, especially when workloads and resource limits are constantly changing [19]. Current methods frequently have trouble dealing with the intricacies of networked cloud infrastructures, where the dependency of several nodes and changing traffic conditions make load balancing difficult. This study suggests that combining the Rock Hyrax strategy with Regret Minimization strategies could lead to a better solution by reducing the regrets people have about their decisions in real-time load-balancing situations [22]. The goal of this study is to create a new optimization framework that can adjust to the changing nature of cloud networks. This will make better use of resources and improve the overall performance of the system [20, 21]. However, it still leaves room for further exploration, especially in quantifying the impact of factors such as technological readiness, top management support, relative advantage, competitive pressure, organizational resistance, system complexity, and data security [23, 24, 31]. Table I shows a critical literature analysis.

TABLE I. Critical Literature Analysis

Study /Approach	Contribution	Limitation / Gap Identified
Dynamic Load Balancing [17,18]	Improves efficiency, reduces overhead	Struggles under highly dynamic workloads and distributed nodes
Nature-inspired Metaheuristics [19]	Flexible, adaptive solutions	Often computationally expensive, lacks real-time responsiveness
Regret Minimization [22]	Handles decision-making under uncertainty	Not yet applied in large-scale cloud balancing contexts
Cloud Performance Factors [23,24,31]	Identify organizational & technical influences (e.g., security, complexity)	Mostly conceptual, not integrated into optimization algorithms

B. RHSO

Many different methods have been created to make the most of energy use in cloud computing systems, but a lot of them still have trouble managing load distribution well in a changing and unpredictable environment [32]. Researchers have come up with hybrid ways to use less energy, but they don't completely solve the problems of uncertainty and load changes in cloud computing [25]. When working with data that changes a lot and is hard to understand, like loads that change or are hard to estimate, these methods can run into problems. This shows that only optimizing energy isn't enough to get the job done without taking into account how the load changes in the system. Also, the Rock Hyrax Swarm Optimization (RHSO) algorithm that was created works well for selecting features for credit card fraud detection systems, but it is still not very useful in the cloud computing world [26]. These methods are more often used for classification and pattern recognition tasks, and they don't always take into account how uncertain it is to manage resources in the cloud. We still need to look into how well RHSO can optimize load distribution in real time in a cloud computing environment with different resources and needs that aren't always clear [27, 28]. There are many different load optimization models, such the Random Regret Minimization (RRM) model. However, it still has problems when it comes to

being used in the cloud computing environment, which is very dynamic and uncertain [29]. On the other hand, Bayesian Optimization with expectation correction for cumulative regret reduction and other similar methods are more focused on finding the best solution in a noisy environment [30].

This study's goal is to fill in the gaps by creating a new method that uses both the Rock Hyrax Swarm Optimization algorithm and the Regret Minimization strategy to optimize load balancing in cloud computing networks. One of the main goals of this research is to find a solution to the problem of uncertainty in load distribution that cloud computing systems confront. This is because they often need to allocate resources fast and accurately when demand changes. The system can lower losses from bad judgments, make load sharing more efficient, and use less energy more effectively by utilizing a Regret Minimization strategy based on modularity optimization [13]. This study focuses to improve the Rock Hyrax and Regret Minimization methods for optimizing load balancing on networked cloud-based computing systems. The goal of this study is to find and improve optimization methods that use algorithms to make cloud computing load allocation decisions more accurate. One of the problems with load-balancing is how to deal with changing and different loads without affecting performance or resource availability. The idea behind this study is that using Rock Hyrax with Regret Minimization will give better results than traditional load-balancing approaches.

C. Research Stages

The focus of this research is to identify problems associated with throughput optimization to produce an optimal transfer rate of data delivery services. The stages used in this research are developed from previous research. Making of the modelling algorithm is given in the following Fig. 1:

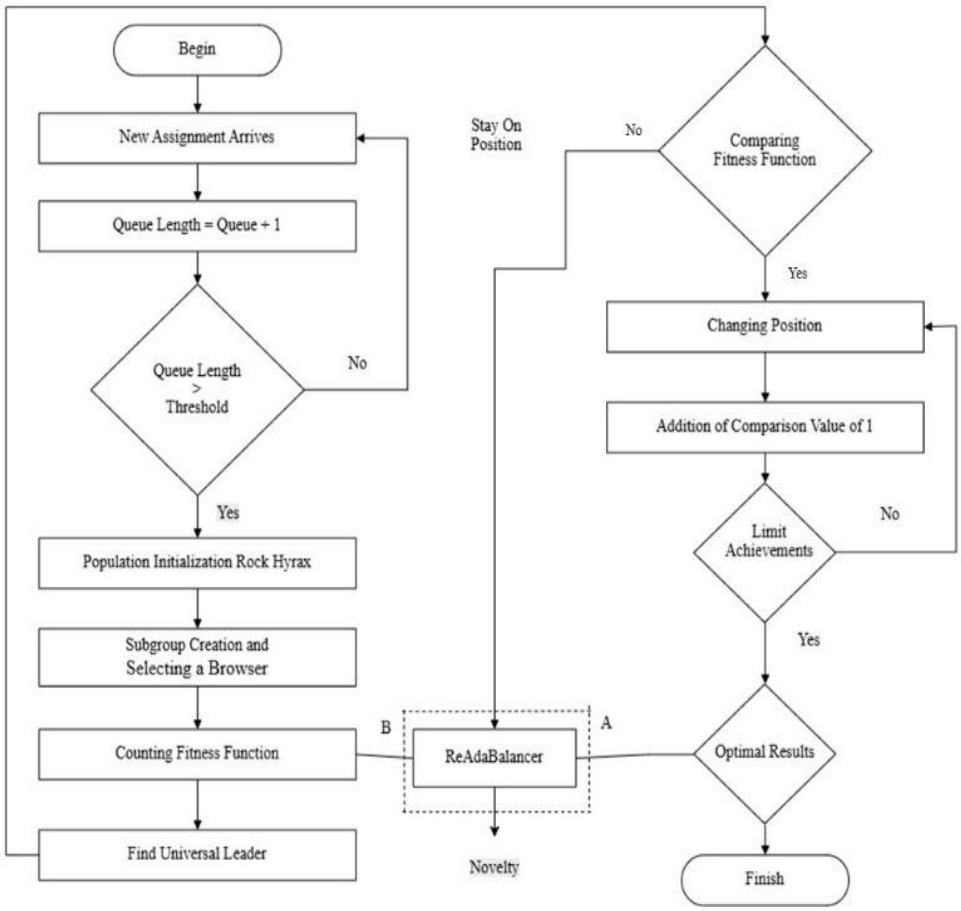


Fig. 1. Process algorithm.

The Proposed Rock Hyrax Load Balancing Algorithm is intended to efficiently manage workloads in a cloud computing environment. The following is a complete process flow story of how the algorithm functions:

1) Job submission: Users submit their tasks to the broker, which serves as the central controller for job scheduling and resource allocation in the cloud environment. The broker is also responsible for managing the distribution of tasks across different virtual machines (VMs).

2) Initial load assessment: The algorithm starts by checking the workload and resources of each virtual machine to find virtual machines that are overloaded. This is done by comparing the current load with a preset threshold.

3) Threshold monitoring: The waiting job queue length for each virtual machine (VM) has a threshold. If the queue length exceeds this threshold, the algorithm triggers load-balancing actions to ensure that no single VM becomes a bottleneck due to excessive demand.

4) Dynamic load redistribution: The algorithm evaluates the current state of the virtual machine (VM) when new jobs arrive. If the VM is overloaded, the algorithm reallocates waiting tasks to other VMs with available capacity. Dynamic redistribution and monitoring ensure optimal performance and prevent server failure.

5) Quality of Service (QoS) considerations: The Rock Hyrax algorithm considers quality of service parameters such as maxspan, throughput, response time, and energy efficiency. This ensures that load balancing addresses live workloads and optimizes overall system performance.

6) Continuous feedback loop: The capacity and workload of each server are monitored and sent back to the broker. The broker can make informed decisions about task allocation and adjust load distribution strategies as needed with this feedback loop.

7) Execution and monitoring: Once tasks are allocated to the appropriate virtual machines (VMs), the algorithm continuously monitors the execution of these tasks. If any VM is overloaded again, the process repeats, ensuring that the system remains balanced and efficient during job execution.

Performance Evaluation: Finally, the results show that makespan and energy consumption become lower, which indicates that Rock Hyrax's load balancing algorithm really works well when organizing resource allocation in cloud computing.

D. Preprocessing and Augmentation Data

This study looks at common preprocessing or data augmentation methods that are utilized in machine learning. If we think about the process in terms of data handling and optimization, though, we can see that the system is doing some "preprocessing" processes to manage tasks. For instance, when a new task comes in, the system will add it to the queue and check to see whether it goes over the limit. This is like getting data ready for more processing. This phase makes sure that only jobs that are too heavy are processed, which cuts down on unneeded overhead. Also, setting up the "Rock Hyrax" population and making subgroups can be considered as setting up several possible solutions (or data points) for optimization, like feature selection or data splitting. The system does "data augmentation" by comparing fitness functions and changing the placements of agents based on these comparisons. This is done again and over again. You could say that this is an addition to the original set of possible answers, with each repetition making the solutions better and more suited to the problem at hand. Also, the addition of a comparison value and the constant changes to the agents' positions are like the iterative nature of data augmentation, where new data variations are made by making changes and comparisons, which makes the model better at adapting and finding the best solution. These phases make sure that the optimization process looks at a lot of different options, which makes the system stronger and more efficient, just like how data augmentation makes models more general.

E. Model Architecture

The model architecture refers to the overall structure of a machine learning model or computational model, which defines the components, their relationships, and the flow of data through the system. The model architecture serves to design, build, and understand how the various parts of the model interact to achieve the desired functionality.

The following is an architectural diagram of the M/M/s/k queue pseudocode based on the pseudocode that has been made. This diagram illustrates how customers enter the system, wait in the queue if needed, are served by the server, and then exit the system or are rejected if the queue is full, as can be seen from Fig. 2:

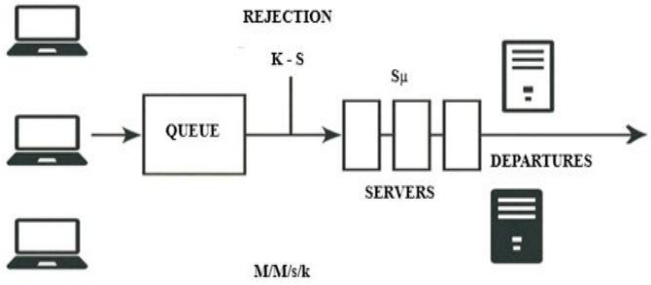


Fig. 2. Architectural diagram of the M/M/s/k queuing model.

Fig. 2 shows a queuing system in a crowded place, such as a fast food restaurant or a call centre. Randomly come to the queue of customers (represented by a computer icon). However, this queue has a limit that not everyone can join. If the queue is full (exceeding k capacity), you will be denied entry. In this system, several servers (s) serve customers one by one, and each server has a certain service speed (μ). Once served, exit the system. This model is called M/M/s/k, and is often used to measure the performance of capacity-constrained systems, such as how fast the service is, and how busy the system itself is.

Load balancers distribute load based on server capacity and user demand. Basic formula:

$$S_i = argarg\left(C\left(S_j\right) + L\left(S_j\right)\right), \quad (1)$$

where, *Si* is the selected server;

Sj is each server in the pool;

C(*Sj*) is the processing capacity of server j;

L(*Sj*) is the latency of server j.

The load balancer selects the server with the lowest C+L value.

1) ReAdaBalancer algorithm: ReaAda balances tasks with the smallest load and considers redistribution. Determining the smallest load on a server can be analogous to a mathematical formula that describes the probability of a server being active or not, based on the load received. With a value, which is the size of the load, and other parameters, such as α and β, which serve to adjust the sensitivity and threshold. Creating the server inactive is significantly larger, creating an efficient that when the server does not receive many requests.

$$P(S_i) = \frac{1}{1 + e^{-(\alpha U_i - \beta)}}, \quad (2)$$

where, $P(S_i)$ is the probability of selecting server i;

U_i is the utilization of server i;

α and β are parameters that control the sensitivity of the algorithm to server load. The lower the U_i value, the greater the chance of that server being selected.

2) Regret minimization: Regret Minimization aims to minimize the difference between the best decision and the decision taken:

$$R_t = \sum_{t=1}^{T} w_t (O_t - A_t), \quad (3)$$

where, R_t is the regret at time t;

O_t is the optimal outcome that should be obtained if the best decision is taken;

A_t is the result obtained from the decision taken by the system;

W_t is the weight that determines how important the error is at time t.

The goal of this algorithm is to make $R_t \rightarrow 0$ over time by adaptive learning.

3) Performance evaluation: Performance evaluation is calculated by looking at throughput (T), response time (R), and error rate (E):

$$K = \frac{T}{R + E}, \quad (4)$$

where, K is the performance score;

T is the number of successfully processed requests;

R is the average response time;

E is the number of errors that occur.

The higher the K, the better the system performance.

Based on the idea of queuing theory, the M/M/k Load Balancer Architecture is arranged here in an organized table. Typically used in high-performance distributed systems where multiple servers (k) handle randomly arriving jobs (Poisson process) and each server offers exponential service time. Describes the architectural structure of the M/M/k queue system with a Load Balancer. These system requests arrive at a rate of λ and are then distributed by the Load Balancer to the k available servers. If all servers are busy, they enter the Queue before being processed. Utilization (ρ) is an important parameter that determines whether or not the system is in a stable state. There are two main probabilities: P_0, which indicates that there are no customers in the system, and Pq, which indicates the chance that customers will have to queue before being served.

Output based on system performance is rather different. While L (System Length) shows the total customers in the system, including those being served. Shows L (Queue Length). In terms of time, Wq (Queue Wait Times) calculates how long the average customer has to wait in the queue, while W (System Wait Times) is the total time spent by the system. After proposals, requests go out as completed requests.

F. Simulation Model

A simulation model is a computational representation of a process, system used to simulate under various conditions. This model is built into a diagrammatic architecture.

Here is a diagram of the queue simulation architecture based on the pseudocode you provided. This diagram shows the relationship between the main components, such as customer, simEventList, queue, and queueSim, which can be seen from Fig. 3.

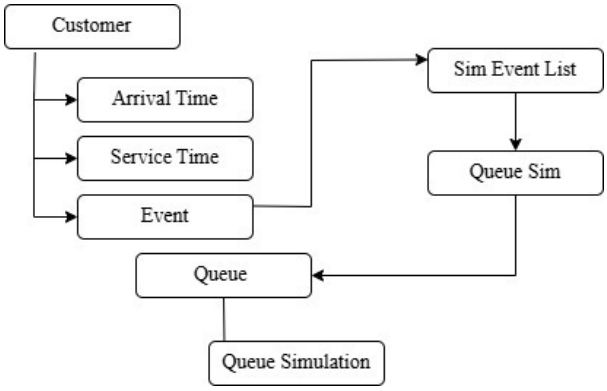


Fig. 3. Architectural diagram of the queue simulation model.

Performance evaluation at the final stage enables continuous monitoring and improvement of the system to remain optimum in handling demand. This approach is not operational, but it also minimizes the potential for errors and quality degradation.

G. Experimental Setup

Experimental setup refers to the arrangement of equipment, in terms of both hardware and software, to analyze the phenomenon under study.

- The hardware and software used for this study are:
  - The research environment (CloudSim, iFogSim);
  - Infrastructure for both private and public clouds (OpenStack, AWS, Azure, GCP);
  - Microservices and container platforms like Kubernetes and Docker;
  - systems for edge and fog computing;
- Frameworks for serverless computing.

Adaptive and decentralized load optimization is its best feature. This makes it very useful in multi cloud, hybrid, and edge-cloud environments where real-time decision-making and self-learning are very important.

III. Results and Discussion

This study show that the integration of Rock Hyrax and

Regret Minimization for cloud computing optimization with the ReAdaBalancer concept keeps the system stable and getting data quickly is very important. This is done by optimizing resource allocation. Under the same conditions, ReAdaBalancer cuts the response time by about 67% and the request rejection rate by more than 50% compared to the standard approach. This is done by adjusting to the traffic in real time and considering the server status. Unlike previous works such as [22], where only using the Rock Hyrax algorithm shows lower accuracy due to its inability to stress the system remains stable in optimizing resource allocation. In contrast to [28], where they propose a near-optimal discrete optimization approach for experimental design using a Regret Minimization framework. This approach aims to select the statistically most efficient subset of design points from a larger data set. In addition, while [29] this RRM model can be adapted to plan resource allocation by considering regret over previous decisions.

A. *Maths Pseudocode*

Between mathematical conceptual thinking and effective programming implementation, Maths Pseudocode is a logical and methodical presentation of mathematical algorithms organized in half-code form. including the following:

#Mathematical pseudocode

import maths

def lambdaN(lam, k, n):

Fig. 4 is the pseudocode of the simulation results using Python.

```
if n < k:
return lam
return 0
def muN(mu, s, n):
if n < s:
return n*mu
return s*mu
def P0(lam, mu, s, k):
sum = 1.0
for i in range(1, s):
sum += (pow((lam/mu), i) / math.factorial(i))
sum  += ((pow((lam/mu),  s)  /  math.factorial(s))) * ((1-pow((lam/(s*mu)), (k-
s+1)))) / (1-(lam/(s*mu)))))
 return 1/sum
```

Fig. 4. Arrival function, service level, and probability.

Summary of Fig. 4

The code above implements the M/M/s/k queuing system, which analyses a system with a limited number of servers (s) and a maximum capacity (k). The lambdaN function stops if the capacity is full, which stops the arrival of customers (λ). Calculates the service level (μ), which increases with the number of customers but is limited by the number of servers. The P0 function terminates an empty system (P_0), which is useful for evaluating queuing systems. This model is useful for others with capacity constraints, such as call centers, computer servers, or service systems.

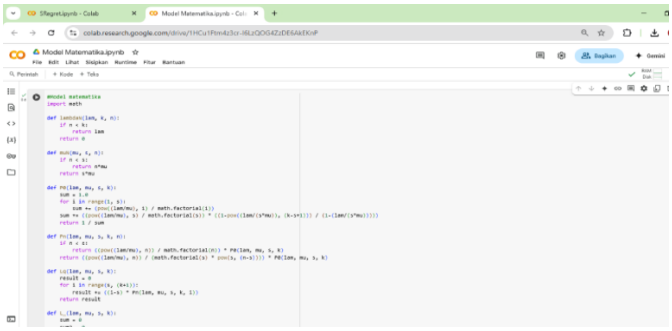


Fig. 5. Running a mathematical model.

Fig. 5 shows a snippet of Python code written on Google Colab, titled Mathematical Models. This script implements several complex mathematical functions used for probabilistic modelling or queuing theory, marked with common notations such as λ (arrival rate), μ (service rate), and k (number of customers in the system). The functions P0 and Pn indicate attempts to calculate the probability of states in a queuing system, while the functions PC and PQ indicate probability calculations in the context of a limited system capacity or queue. The use of math.factorial, math.pow, and the use of nested loops reflect typical combinatorial and exponential calculations in the Poisson or M/M/1/K models.

B. *Simulation Pseudocode*

Run through simulations, before they are used in actual computer code, pseudocode, a methodical description of a logically ordered simulation process, allows the systematic construction and analysis of scenarios, therefore modelling the dynamics of complex systems and enabling the comprised. The following Fig. 6 shows the customer function, simEventList..

```
#Simulation pseudecode
import numpy as np
import random
import maths
import queue
class customer:
 departTime = 0
 def __init__(self, at, st, s):
        self.arrivalTime = at
        self.serviceTime = st
        self.event = s
class simEventList:
 def __init__(self, lam, mu, n = 0):
        tempServiceTime = int(np.random.exponential(1/mu)*60)
        self.list = [customer(0, tempServiceTime, 'arrival')]
        for i in range(n-1):
                tempServiceTime = int(np.random.exponential(1/mu)*60)
                tempInterArrivalTime = int(np.random.exponential(1/lam)*60)
```

Fig. 6. Customer function, simEventList .

Summary of Fig. 6

Part of a queuing system simulation that uses an exponential distribution to determine customer arrival and service times. Customer The class represents a customer with attributes of arrival time, service time, and event type (or departure). Meanwhile, the simEventList class manages the list of events in the simulation, generates service times, and determines the inter-arrival time of customers based on an exponential

distribution. This simulation is useful for analyzing the performance of other systems that have limited capacity, with queues, such as contact centers, banks, or other systems.

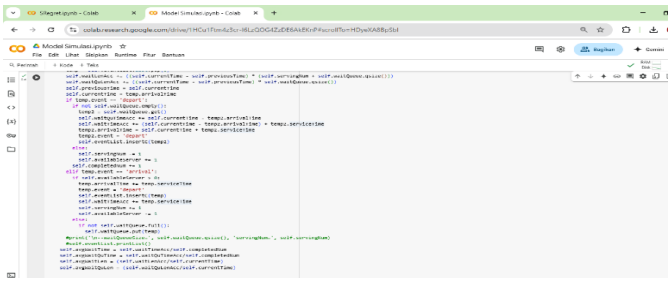


Fig. 7. Running system simulation.

Summary of Fig. 7

Simulation of a queuing system that uses an exponential distribution to determine customer arrival and service times. Each customer has attributes of arrivalTime, serviceTime, and event type. In this simulation, events are in the simEventList class, which ensures that each customer's arrival and departure are sorted chronologically. The available servers, as well as calculating performance metrics such as average time and queue length, the queueSim class is responsible for running the simulation by maintaining the queue. If all servers are busy, customers will be put into a waiting queue, provided there is still available capacity.

The simulation is run in a loop that processes each event in the list sequentially. If a customer arrives and there is a server, they are served immediately. Otherwise, they enter the queue if there is still space. Once in the queue, they begin to be served, the server becomes available again, and the next customer follows. The final results of the simulation include various statistical measures that can be used to evaluate the efficiency of the queuing system, such as average waiting time and queue length. The model can be used in various real-world applications, such as customer service, traffic management, and data processing systems.

C. Load Balancer System Test Pseudocode

The Load Balancer System Test Pseudocode is designed to confirm the fair and efficient traffic distribution between servers by logical simulation before technical implementation is carried out. Pseudocode is a methodical depiction of the load balancer system test flow comprising the following, as shown in Fig. 8.

Summary of Fig. 8

The load balancing () function simulates the M/M/k queue model with multiple servers. It calculates system utilization, estimated number of customers in the system and queue and arrival rate (λ), service rate (μ), and arrival rate (λ). With this simulation, it is possible to evaluate service efficiency in various queuing systems, such as call centres, hospitals, and data centres, in order to optimize capacity and reduce customer waiting time.

Fig. 9 shows the running of simulation results using Python.

```
import numpy as np
import matplotlib.pyplot as plt
def loadBalancer(lam, mu, k, s=5):
 """
Simple simulation of the M/M/k queue model.
lam: arrival rate (lambda)
mu: service rate (mu)
k: number of servers
s: number of lambda simulation points
Returns:
x -> lambda array
simLy, Ly -> number of customers in the system (simulation vs theory)
simWy, Wy -> customer waiting time in the system (simulation vs theory
simLqy, Lqy -> number of customers in queue (simulation vs theory
simWqy, Wqy -> waiting time in queue (simulation vs theory)
 """
x = np.linspace(lam, lam + 1, s) # Simulating multiple lambda values
rho = x / (k * mu) # System utilisation
# M/M/k queue system estimation formula
Ly = rho / (1 - rho) # Estimated number of customers in the system
Wy = Ly/x # Estimated total waiting time
Lqy = Ly - (x/mu) # Estimated number of customers in the queue
Wqy = Lqy / x # Estimated waiting time in queue
```

Fig. 8. Load balancer function.

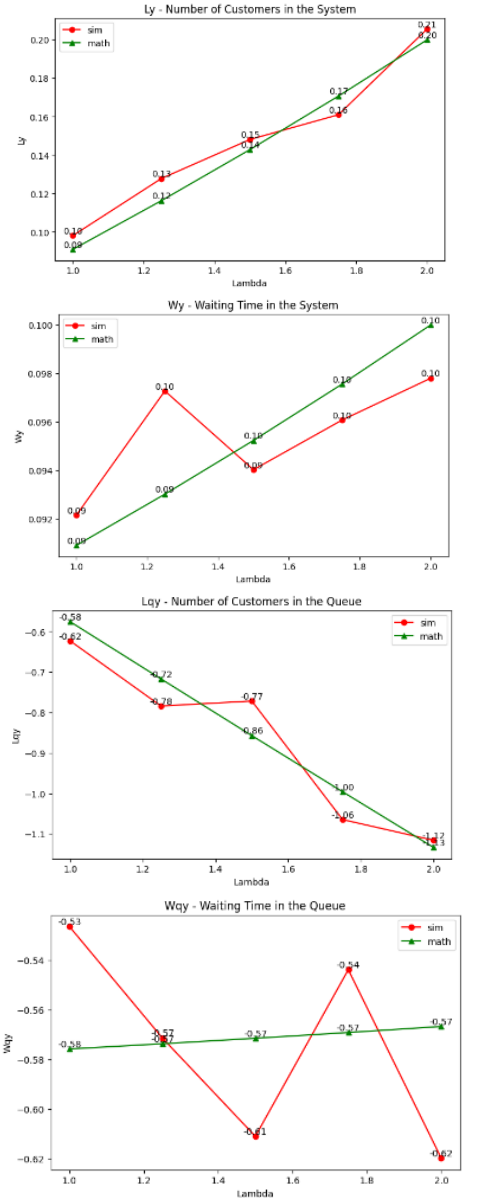


Fig. 9. Running simulation test of load balancer system.

Simulation summary of Load Balancer Test Model

Load balancing, with the code above, simulates a multiple server M/M/k queue system. The load balancing function calculates queue parameters such as the number of customers in the system (Ly), the waiting time of customers in the system (Wy), the number of customers in the queue (Lqy), and the waiting time in the queue (Wqy). These results are calculated both theoretically and through simulations of varying the value of λ (arrival rate) within a certain range. The simulation is added with random noise in the real system to represent in. After the calculation is complete, the results are visualized using Hutmatplotlib to compare the theoretical values and simulation results.

However, there are some implementations of this. Ly is not suitable for calculating Ly due to multiple servers (M/M/k), as it only applies to M/M/1 systems. Not checking the system, which should ensure that the utilization ratio $\rho = \lambda/(k\mu)1$ in order to keep the queue under control. If $\rho > 1$, the system is not able to increase indefinitely. For improvement, it is necessary to use the Erlang-C formula to get a more accurate estimation in the M/M/k model.

D. Innovative ReaAdaBalancer

An innovative load balancing method called RegretAdaptive Load Balancing (ReAdaBalancer) dynamically optimizes demand distribution by means of regret reduction. The simulation in Table II shows how helpful this approach is in settings with varying workloads and heterogeneous servers.

TABLE II. REGRETADAPTIVE LOAD BALANCER PERFORMANCE RESULTS

Server ID	Final Load	Final Regret	Utilization (%)
0	11	0	11.0%
1	5	0	5.0%
2	0	0	0.0%
3	0	0	0.0%
4	6	0	6.0%

Table Analysis:

1) Final Load: Shows the final load amount that each server has after 20 iterations.

2) Final Regret: None of the servers are overloaded, so the regret score remains 0.

3) Utilization (%): All servers have a low utilization rate, which means the load has been distributed fairly well.

Fig. 10 shows the running of simulation results using Python.

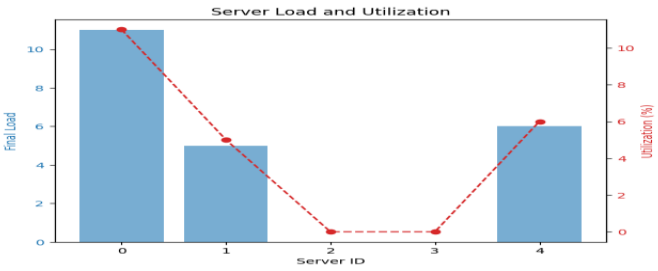


Fig. 10. Simulation graph of adaptive load balancer regret.

This graph presents "Server Load and Utilization" data related to the final load and utilization rate of five server units identified by IDs 0 to 4. The observation shows that there is a significant imbalance in the workload distribution between servers. The server with ID 0 had the highest final load of 11 and a utilization rate of 10%, indicating that it was working intensively and was likely to be the main center of data processing. Meanwhile, server ID 1 showed a drastic drop with an end load of only 5 and a utilization rate of 5%, suggesting a reduction in activity or an imbalance in load allocation. More strikingly, server IDs 2 and 3 recorded zero values on both metrics, signifying that they were not involved in any computing processes at all, potentially indicating high idle time or no assignment at all. In contrast, server ID 4 shows an increase again with a final load of 6 and utilization of 6 %, indicating moderate involvement in system activity. Overall, this graph reflects an uneven load distribution among the server units. This imbalance can have an impact on operational efficiency as well as hardware lifespan. Therefore, there is a need to evaluate the scheduling and load distribution strategies to ensure that all system resources are utilized optimally and proportionally.

E. Discussions

In summary, the discussion confirmed that the proposed ReAdaBalancer, which includes preprocessing as an important first step to prepare the data for the load balancing [10] process, and augmentation, which improves the model in the face of dynamic changes in cloud environments, both help provide smart and efficient solutions for load optimization that can be used on a large scale and in cloud environments that change quickly.

Despite its advantages, this research has some limitations, including that the algorithm becomes more difficult to compute as the number of nodes and the size of the cloud network increase. The algorithm may have difficulty in making timely adjustments when there is a sudden change in load or system failure, which can lead to poor conclusions. Future improvements may include: 1) the use of parallel computing and distributed processing to speed up the performance of the algorithm, 2) the use of online learning or incremental learning techniques to keep the model up-to-date as new data comes in, 3) the addition of reinforcement learning for parameter adjustment to provide a dynamic solution that avoids incorrect parameter selection, 4) the addition of distributed error detection and self-healing features to the algorithm can make it more fault-tolerant. For example, replication techniques can be used to copy the load to a backup server in case of a failure, and 5) the use of hybrid algorithms that combine Rock Hyrax with other faster optimization methods such as shortest path algorithms or approximate algorithms can help you make decisions faster and ensure that load distribution occurs quickly and efficiently for real-time applications.

ReAdaBalancer has been shown to work very well for improving system resilience and availability. Helping to make better use of resources and adapt to changing conditions. The Rock Hyrax Approach and Regret Minimization will focus on making cloud computing [13] more scalable, able to adapt automatically, able to handle failures, better at managing energy, and more secure in the future. This research can make

a cloud system that is more efficient, safe, and able to adapt to future issues by looking into how to combine emerging technologies like machine learning, blockchain, and real time processing and coming up with solutions for apps that need to do more than one thing.

IV. Conclusion

The model that combines Rock Hyrax and Regret Minimization for cloud computing optimization with the ReAdaBalancer concept keeps the system stable, and the ReAdaBalancer cuts response time by about 67% and request rejection rate by more than 50% compared to the standard approach by adjusting traffic in real-time.

Test Server with ID 0 has the largest final load of 11 and utilization rate of 10%, which means that it is working hard and is probably the major place where data is processed. Server ID 1, on the other hand, has a final load of only 5 and a utilization rate of 5%, which means that either activity has decreased or load allocation has become unbalanced. This performance boost can be a clever and effective way to optimize load that works on a large scale and in cloud environments that change quickly.

Results are in line with the goal and unique features. The suggested load balancing optimization method effectively solves the main problems found in the study, such as being able to adapt to changing system conditions, using a lot of energy, and not being very good at distributing workloads in real time. Rock Hyrax and Regret Minimization work together to give a unique advantage over standard algorithms by changing to traffic patterns in real-time. This makes the system more responsive and uses resources better.

References

[1] Bello, Sururah A., et al. "Cloud computing in construction industry: Use cases, benefits and challenges." Automation in Construction 122 (2021): 103441. Available at: https://doi.org/10.1016/j.autcon.2020.103441.

[2] Shukur, H. et al. (2020) ‘Cloud Computing Virtualization of Resources Allocation for Distributed Systems’, Journal of Applied Science and Technology Trends, 1(2), pp. 98–105. Available at: https://doi.org/10.38094/jastt1331.

[3] Kirtirajsinh Zala et al., (2022), PRMS: Design and Development of Patients' E-Healthcare Records Management System for Privacy Preservation in Third Party Cloud Platforms, https://doi.org/ 10.1109/ACCESS.2022.3198094: IEEE Access. (2022): 85777 - 85791.

[4] Khare, Shivangi, Uday Chourasia, and Anjna Jayant Deen. "Load balancing in cloud computing." Proceedings of the International Conference on Cognitive and Intelligent Computing: ICCIC 2021, Volume 1. Singapore: Springer Nature Singapore, (2022): 978-981. Available at: https://doi.org/10.1007/978-981-19-2350-0_58.

[5] Shafiq, Dalia Abdulkareem, N. Z. Jhanjhi, and Azween Abdullah. "Load balancing techniques in cloud computing environment: A review." Journal of King Saud University-Computer and Information Sciences 34.7 (2022): 3910-3933. Available at: https://doi.org/10.1016/j.jksuci.2021.02.007.

[6] Mishra, Sambit Kumar, Bibhudatta Sahoo, and Priti Paramita Parida. "Load balancing in cloud computing: a big picture." Journal of King Saud University-Computer and Information Sciences 32.2 (2020): 149-158. Available at: https://doi.org/10.1016/j.jksuci.2018.01.003.

[7] Jena, Uttam Kumar, Pradipta Kumar Das, and Manas Ranjan Kabat. "Hybridisation of meta-heuristic algorithm for load balancing in cloud computing environment." Journal of King Saud University-Computer and Information Sciences 34.6 (2022): 2332-2342. Available at: https://doi.org/10.1016/j.jksuci.2021.02.004.

[8] Singh, Abhilash, Sandeep Sharma, and Jitendra Singh. "Nature-inspired algorithms for wireless sensor networks: A comprehensive survey." Computer Science Review 39 (2021): 100342. Available at: https://doi.org/10.1016/j.cosrev.2020.100342.

[9] Katangur, Ajay, Somasheker Akkaladevi, and Sadiskumar Vivekanandhan. "Priority weighted round robin algorithm for load balancing in the cloud." 2022 IEEE 7th international conference on smart cloud (SmartCloud). IEEE, (2022): 5179. Available at: https://doi.org/10.1109/SmartCloud57250.2022.00104.

[10] Jaganathan, Subash Chandra Bose, Ramesh Saha, and S. Kannadhasan. "An Efficient Enhanced Dynamic Load Balancing Weighted Round Robin Algorithm for Virtual Machine in Cloud Computing." Journal of Algebraic Statistics 13.2 (2022): 2121-2128. Available at: https://doi.org/10.17762/jas.v13i2.394.

[11] Nakas, Christos, Dionisis Kandris, and Georgios Visvardis. "Energy efficient routing in wireless sensor networks: A comprehensive survey." Algorithms 13.3 (2020): 72. Available at: https://doi.org/10.3390/a13030072.

[12] Kashani, Mostafa Haghi, and Ebrahim Mahdipour. "Load balancing algorithms in fog computing." IEEE Transactions on Services Computing 16.2 (2022): 1505-1521.Available at: https://doi.org/10.1109/TSC.2022.3174475.

[13] Pradhan, Arabinda, Sukant Kishoro Bisoy, and Amardeep Das. "A survey on PSO based meta-heuristic scheduling mechanism in cloud computing environment." Journal of King Saud University-Computer and Information Sciences 34.8 (2022): 4888-4901. Available at: https://doi.org/10.1016/j.jksuci.2021.01.003.

[14] Akhtar, Talha, Najmi Ghani Haider, and Shariq Mahmood Khan. "A comparative study of the application of glowworm swarm optimisation algorithm with other nature-inspired algorithms in the network load balancing problem." Engineering, Technology & Applied Science Research 12.4 (2022): 8777-8784. Available at: https://doi.org/10.48084/etasr.4999.

[15] Breish, Firas, Christian Hamm, and Simone Andresen. "Nature's Load-Bearing Design Principles and Their Application in Engineering: A Review." Biomimetics 9.9 (2024): 545. Avalable at: https://doi.org/10.3390/biomimetics9090545.

[16] Jiao, Licheng, et al. "Nature-Inspired Intelligent Computing: A Comprehensive Survey." Research 7 (2024): 1-38. Available at: https://doi.org/10.34133/research.0442.

[17] Maina Lawan, M., Oduoza, C. and Buckley, K. (2021) ‘A Systematic Review of Cloud Computing Adoption by Organisations’, International Journal of Industrial and Manufacturing Systems Engineering, 6(3), p. 39. Available at: https://doi.org/10.11648/j.ijimse.20210603.11.

[18] Sharmah, D. and Bora, K.C. (2024) ‘A Survey on Dynamic Load Balancing Techniques in Cloud Computing’, Lecture Notes in Electrical Engineering, 1088(2), pp. 273–282. Available at: https://doi.org/10.1007/978-981-99-6855-8_21.

[19] Gupta, R. and Sharma, O.P. (2024) ‘Optimization of Load Balancing in Cloud Computing through Nature- Inspired Metaheuristic Algorithms’, pp. 3216–3226. Available at: https://doi.org/10.52783/jes.8054.

[20] Sakhri, A. et al. (2024) ‘A digital twin-based energy-efficient wireless multimedia sensor network for waterbirds monitoring’, Future Generation Computer Systems, 155(February), pp. 146–163. Available at: https://doi.org/10.1016/j.future.2024.02.011.

[21] Chawla, K. (2024) ‘Reinforcement Learning-Based Adaptive Load Balancing for Dynamic Cloud Environments’. Available at: http://arxiv.org/abs/2409.04896.

[22] Singhal, S. et al. (2024) ‘Energy Efficient Load Balancing Algorithm for Cloud Computing Using Rock Hyrax Optimization’, IEEE Access, 12(February), pp. 48737–48749. Available at: https://doi.org/10.1109/ACCESS.2024.3380159.

[23] Singhal, S. et al. (2023) ‘Energy Aware Load Balancing Framework for Smart Grid Using Cloud and Fog Computing’, Sensors, 23(7). Available at: https://doi.org/10.3390/s23073488.

[24] Yang, P. et al. (no date) ‘Reducing Idleness in Financial Cloud Services via Multi-objective Evolutionary Reinforcement Learning based Load Balancer arXiv : 2305. 03463v2 [cs .NE] 23 Nov 2023’. Available at: https://doi.org/10.1007/s11432-023-3895-3.

[25] Kak, S. M., Agarwal, P., Alam, M. A., & Siddiqui, F. (2024). A hybridized approach for minimizing energy in cloud computing. Cluster Computing, 27(1), 53-70. Available at: https://doi.org/10.1007/s10586-022-03807-9.

[26] Padhi, Bharat Kumar, et al. "RHSOFS: feature selection using the rock hyrax swarm optimisation algorithm for credit card fraud detection system." Sensors 22.23 (2022): 9321. Available at: https://doi.org/10.3390/s22239321.

[27] Jagadesh, B. N., et al. "Segmentation Using the IC2T Model and Classification of Diabetic Retinopathy Using the Rock Hyrax Swarm-Based Coordination Attention Mechanism." IEEE Access (2023): 124441 - 124458. Availavle at: https://doi.org/10.1109/ACCESS.2023.3330436.

[28] Allen-Zhu, Zeyuan, et al. "Near-optimal discrete optimisation for experimental design: A regret Minimization approach." Mathematical Programming 186 (2021): 439-478. Available at: https://doi.org/10.1007/s10107-019-01464-2.

[29] Mengjie L et al., Random Regret Minimization Model for Variable Destination- Oriented Path Planning, https://doi.org/ 10.1109/ACCESS.2020.3021524. IEEE Access (2020). 163646 - 163659.

[30] Hu, Shouri, et al. "Adjusted expected improvement for cumulative regret Minimization in noisy bayesian optimisation." arXiv preprint arXiv:2205.04901 (2022):1:31. Available at: https://doi.org/10.48550/arXiv.2205.04901.

[31] Xia, J., Li, M., & Guo, Y. (2009). The Queuing Model of M/M/S/K+ M Based on the Impatience and Changeable Service Rate. Journal of Systems Science & Information, 7(3). Available at: https://doi.org/10.1007/s10586-022-03807-9.

[32] Cui, T., Yang, R., Fang, C., & Yu, S. (2023). Deep reinforcement learning-based resource allocation for content distribution in IoT-edge-cloud computing environments. Symmetry, 15(1), 217. Available at: https://doi.org/10.3390/sym15010217.

An Augmentation-Based System for Diagnosing COVID-19 Using Deep Learning

Mohamad Shady Alrahhal[1], Mohammad A. Mezher[2],
Osamah A.M. Ghaleb[3], Mohammad Al-Hjouj [4], Raghad Sehly[5], Samir Bataineh[6]
AIMD Laboratory, Computing College, Fahad Bin Sultan University, Tabuk City, KSA[1, 2, 3, 4, 5, 6]
Institute of International Education, SRF program, New York City, United States of America[1]

***Abstract*—Recently, due to the dangerous spread of COVID-19, there has been strong competition among computer science researchers within the scientific research community to employ deep learning for the development of intelligent medical systems that diagnose this illness. Enhancing accuracy is considered the most important objective, and augmentation techniques are used in this context. This study addresses two main issues related to applying augmentation on X-ray and CT-scan images: losing the positional information of augmented medical images and the integration of extracted features while scanning them. The use of the Vision Transformer Structure, supported by a Position-Aware Embedding (PAE) method, is proposed to deal with these issues. Moreover, in this study, a student–teacher-based approach was adopted to enable considerable resistance against training on a small batch of training images. Due to the sensitivity of medical data, preserving the privacy of patients was taken into account by using a pseudonym-based anonymity approach. After evaluations based on accuracy, precision, recall, and specificity metrics, the results showed that the proposed system has a high-level capability to predict class images (X-ray or CT-scan) as well as considerable resistance against training on small medical images.**

***Keywords*—COVID-19; medical images; augmentation; vision transformer; training data ratio**

I. Introduction

The medical sector is considered the most important sector in people's lives because staying healthy leads to higher work productivity and increased happiness. Therefore, governments always rank fighting epidemics at the top of their priorities. In 2019, strange medical signals were observed in some patients in Wuhan, China. These signals showed that the patients had been infected with a serious new strain of coronavirus disease (known as COVID-19) [1, 2]. Subsequently, various parts of the world experienced COVID-19 outbreaks, prompting the World Health Organization (WHO) to inform the public that the COVID-19 crisis had become a global pandemic after the disease was found to have exceptional spreadability. Governments all over the world colluded to fight against the dangerous spread of COVID-19 and its manifold consequences that came in different forms (losses of human life, economic recessions, faltering medical health systems, and necessary lifestyle changes). Most scientists, physicians, and politicians have agreed that the best way to stop the spread of COVID-19 is early diagnosis and detection and adherence to preventive measures such as social distancing [3]. Fig. 1 summarises and arranges the historical events outlined above.

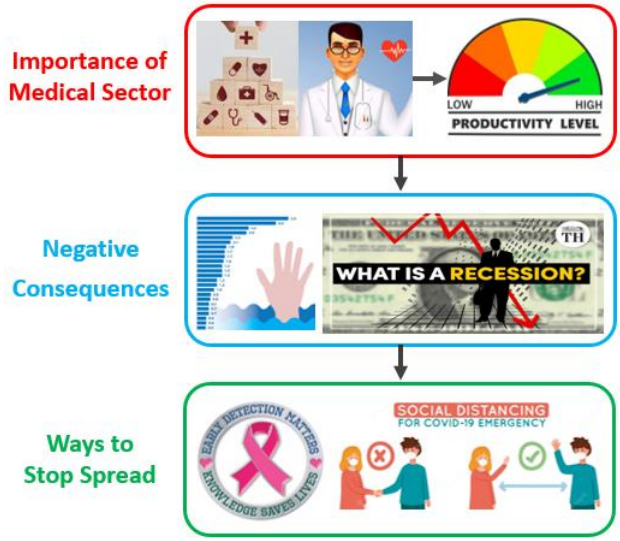


Fig. 1. Historical events resulting from the outbreak of COVID-19.

A. Motivation

The traditional ways of detecting COVID-19 are antibody testing and Quantitative Reverse Transcription Polymerase Chain Reaction (QRT-PCR), both of which have some issues. Antibody testing generates high false-negative rates when it comes to dealing with early active infections [4, 5]. With QRT-PCR, generating results is time-consuming; thus, it is unsuitable when working under severe time constraints. Relying on medical image analysis and making diagnoses via the use of CT images has proven effective in the early detection of COVID-19. However, techniques based on CT images are time-consuming, as they require some pre-activities to be carried out (e.g., transferring patients to a CT facility and the sterilization of medical devices) and other post-activities (e.g., discussing with consultants to analyse the results of the test). Using X-ray images is preferable to the use of CT scan images in terms of cost. However, some research has shown that when COVID-19 progresses in its infection of patients' bodies, some visual features of X-ray images become blurred (i.e., less informative) [6]. As a result, there is a strong motivation among researchers to develop advanced diagnostic systems to detect COVID-19, and here, deep learning (DL) techniques come to the fore. Fig. 2 summarizes this motivation.

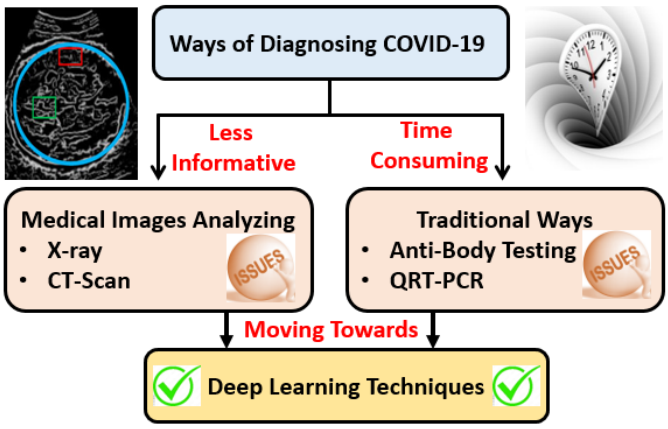


Fig. 2. Motivation.

B. Statement of the Problem

In order to fight against COVID-19, researchers in the field of computer science have made invaluable contributions to the development of intelligent diagnosing systems by enhancing the accuracy of the intelligent systems developed for detecting COVID-19. They have employed the principles of Medical Image Analysis (MIA) and deep learning (DL) to train intelligent systems. X-rays and CT scans are considered the most common sources of medical images. Among the intelligent systems used for diagnosing COVID-19, their low accuracy is the root cause of why many of these systems fail, as insufficient accuracy leads to an increase in the rate of deaths among infected people because of the high number of cases that are falsely classified as negative [7]. Fig. 3 illustrates the low accuracy problem.

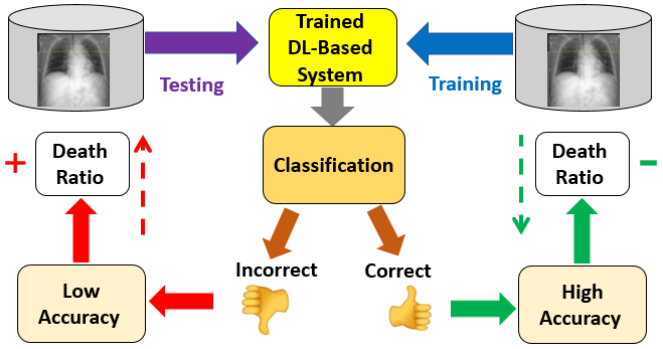


Fig. 3. The low accuracy problem leads to a high death ratio.

One possible way to enhance accuracy is to carry out effective data pre-processing before the system's training stage. In this context, augmentation-based techniques, which center around rescaling medical images to enable the extraction of learning features in an efficient way, are widely used [8]. However, the use of augmentation-based techniques can lead to some issues, including the following: 1) the loss of positional information because the original input medical image is divided into segments, and incorrectly re-ordering segments leads to confusion in the learning process [9]; 2) the integration of extracted features across the entire medical image is crucial to ensure learning on all features [10]; and 3) resistance against different sizes of training datasets becomes a barrier for ensuring that the diagnosing system can be applied in real-time [11]. Fig. 4 illustrates the issues that can arise when using augmentation-based techniques.

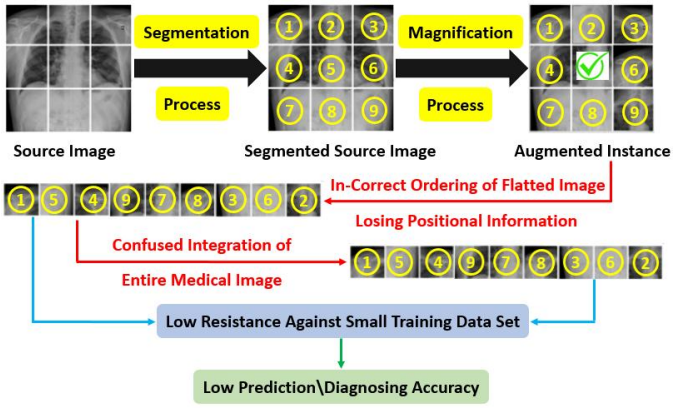


Fig. 4. Issues resulting from poor augmentation.

C. Research Questions

This study aims to address the following research questions:

1) How can one avoid losing the positional information of augmented medical images that are used as inputs for the training process?

2) How can one ensure the integration of extracted features while scanning whole medical images?

3) How can one enable resistant real-time diagnosing against small training datasets?

4) How can the preservation of the privacy of patients, which is a mandatory requirement in the medical sector, be ensured?

D. Contributions

In addressing the research questions listed above, this work aims to make the following contributions to the literature:

- Regarding the first research question, a Position-Aware Embedding (PAE) method that takes into account the corresponding positional information during the flattening process is presented.
- Regarding the second research question, an Integration-based Vision Transformer Structure (IbVTS) is proposed. The self-attention mechanism provided by this VTS, which ensures the integration of the information extracted from each segment of the flattened medical images, is explained.
- Regarding the third research question, the IbVTS is supported by a distillation tactic to activate less training data for the purpose of supporting the classification layer.
- Regarding the fourth research question, a pseudonym-based anonymity approach (PAA) is proposed to hide the personal information of patients.

E. Structure of Study

The rest of this study is structured as follows: Section II outlines related works. In Section III, the architecture of the

proposed intelligent diagnosing system is presented in detail, along with the role of each component. The experimental results are documented in Section IV, and finally, Section V concludes the study and lists directions for future research.

II. Related Work

Many survey-based papers have addressed intelligent systems used to diagnose COVID-19 and provided various taxonomies, such as [12, 13, 14]. The authors of [12] relied on the application of different deep learning approaches used to diagnose COVID-19 (convolutional neural network, Generative Adversarial Network, and Long Short-Term Memory) to derive their taxonomy. The author of [13] presented a taxonomy based on methods used for the data pre-processing stage and techniques used for feature extraction. Convolutional neural networks, recurrent neural networks, deep belief networks, and reinforcement learning are the deep learning architectures used to shape the base of the taxonomy presented in [14]. This work groups intelligent systems into five categories based on the type of learning, as shown in Fig. 5.

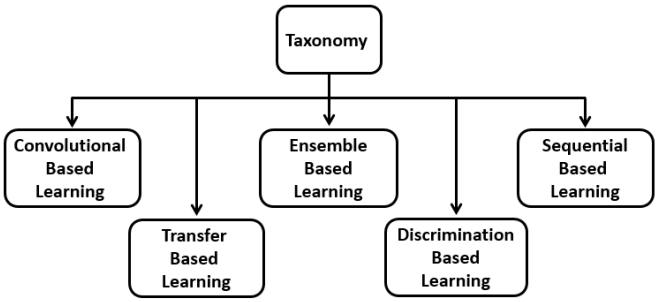


Fig. 5. Taxonomy of intelligent systems used to diagnose COVID-19.

A. Convolutional-Based Learning

In this category, convolutional neural networks dominate the architectures of the proposed systems. CNNs consist of two main types of layers: a convolutional layer and a pooling layer. CNNs employ filtering principles used in the domain of image processing to extract features.

In [15], the researchers proposed an automatic system for diagnosing COVID-19 using a pre-trained U-net architecture. The purpose of this U-net architecture is to segment medical images of the lungs. Next, the segments are used as inputs for a deep neural network to output the class of the given image. They used a dataset of 540 patients, with 313 (138 males; 175 females) of these patients being COVID-19 positive and 229 (88 males; 141 females) being COVID-19-negative. The experiments were conducted on the CT-scan dataset and produced 90.1 % accuracy on binary classes. The CNN-based architecture presented in [16] consists of 25 layers (19 layers for convolution and 6 max-pooling layers). Convolutional layers are connected to batch normalization for input standardization and to regulate the intelligent model. To ensure the continuity of neurons, a leaky rectified linear unit is utilized. The maxpool function is used for the down-sampling of the inputs. The authors used an X-ray dataset comprising 600 medical images for binary classification. An accuracy of 87 % was obtained.

B. Transfer-Based Learning

In this category, two main domains are involved: the source domain and the target domain. The source domain is characterized by models that are data-rich and have high-quality feature extraction capabilities. In contrast, the target domain suffers from low-quality feature extraction capabilities and requires help from pre-trained models. The purpose of transfer-based learning is to shift the knowledge from the source domain to the target domain. The degree of maturity of the transferred knowledge depends on the degree of similarity between the source and target.

Transfer-based learning was applied to CT scan medical images in [17] to avoid the high costs and computational complexity associated with constructing a CNN model. Two main DL-based algorithms were used (ResNet18 and ResNet50) as pre-trained models. Then, discriminant correlation was employed for feature fusion to generate a better image representation. The data of the training dataset were gathered from a local medical center, with there being 420 samples for each class (normal and infected). The results showed that an accuracy of 96.35 % was achieved. The authors of [18] chose to use VGG16, VGG19, ResNet, DenseNet, and InceptionV3 as pre-trained DL-based models with rich knowledge. These architectures were used to transfer information derived from chest X-ray images. To enhance the classification accuracy, a rotation-based augmentation technique was employed and referred to as a heading model. The results obtained after conducting experiments showed that among the five models used, VGG19 achieved the best accuracy (80%).

C. Ensemble-Based Learning

The key idea of ensemble learning is to combine multiple models (referred to as learners) to generate an enhanced model. There is a wide spectrum of tactics used to perform a combination, such as majority voting, plurality voting, and weighted voting.

The authors of [19] proposed an intelligent model that combines EfficientNet and SE-ResNext to predict one of the three classes that label input images (COVID-19, pneumonia, and normal). They chose to use X-ray medical images and a weighted voting-based combination tactic. The results showed that the proposed model achieved a higher accuracy (about 95%) than that achieved by each combined model separately. In the context of ensemble learning, the authors of [20] utilized a stack of pre-trained deep learning models. They used VGG-16 as base learners, trained with a diverse set of inputs, followed by a logistic regression model, the meta learner, to combine the base learner predictions. This combination tactic relies on a fusion technique to enable the system to predict the class of the medical images. They achieved a high level of accuracy (about 89%).

D. Discrimination-Based Learning

The key idea behind discrimination-based learning is to, as a first step, generate tuples that are similar in features and class to real training data. Then, the resultant tuples and real ones are mixed and act as inputs for the discriminator to be handled under a distinguishing process (i.e., to distinguish the real tuples from generated ones). In the context of discrimination-based learning, Generative Adversarial Networks (GANs) rules supreme.

A GAN-based system was proposed by the authors of [21] to construct synthetic CT scan images of both COVID-19 patients and healthy controls. The construction process was performed during the data augmentation stage. The Whale Optimization Algorithm (WOA) was used to optimize the hyperparameters of the GAN. The model was trained using the SARS-CoV-2 CT-Scan dataset, which consists of 2482 images (1252 COVID-19 cases and 1230 healthy cases). In terms of accuracy, the model obtained an accuracy value of 99.22%. A U-Net-based GAN designed for lung segmentation using chest X-ray images was proposed in [22]. This strategy involved employing a fully convolutional neural network in conjunction with the GAN model to segment images of the lungs in order to make COVID-19 diagnoses.

E. Sequential-Based Learning

In this category, the Long Short-Term Memory (LSTM) dominates because it was originally developed to enhance neural networks by enabling learning from sequential data.

The authors of [23] presented a hybrid intelligent model for diagnosing COVID-19 using an LSTM with a CNN. The development of the model consisted of three steps. First, medical images were enhanced in the pre-processing stage based on the contrast technique. Then, the enhanced images were used as model inputs for the purpose of learning. Finally, the Softmax function was used in the classification layer to distinguish between three classes: COVID-19, normal, and pneumonia. This model was trained on the COVID-19 Radiography dataset, which consists of 1143 images of COVID-19 cases, 1341 images of normal cases, and 1345 images of Pneumonia cases. The authors achieved an accuracy of 98.97 %. The authors of [24] utilized a pre-defined histogram threshold to segment CT scan-derived medical images. For feature extraction, both the Q-Deformed entropy (QDE) and a CNN were used. Finally, the extracted features were fused and used as inputs for the LSTM classifier to predict the classes of the images. The highest classification accuracy value obtained was 99.68%.

III. Proposed System

This section presents the proposed intelligent system. Firstly, the framework of the system is presented, followed by the system's architecture and the role of each component involved in constructing the architecture.

A. Framework of Proposed System

The environment within which the proposed system operates is medical health care centers/hospitals. For a given number of patients (Pat), medical machines capture chest images derived from two types of imaging techniques: X-ray, denoted as (MI_{xr}), and CT scan, denoted as (MI_{ct}). Both types form a real dataset [see Eq. (1)]:

$$RDS = \bigcup_{i=1}^{Pat}\{MI_{xr}^{i}, MI_{ct}^{i}\} \quad (1)$$

Each medical image is modeled as follows [see Eq. (2)]:

$$MI_{type\in(xr,ct)}^{i} = \langle IDp, NAMEp, BODp, SEXp, Dis\rangle | i = 1,2,\dots,Pat \quad (2)$$

where, $IDp, NAMEp, AGEp, SEXp, Dis$ represent the personal information of the patient, specifically their identifier, name, date of birth, sex, and medical description or health status.

In reality, in the usage stage of the proposed system (i.e., after training and testing), $MI_{type\in(xr,ct)}^{i}$ is used as an input to predict one of three classes (healthy, pneumonia, or COVID-19), as shown in Fig. 6.

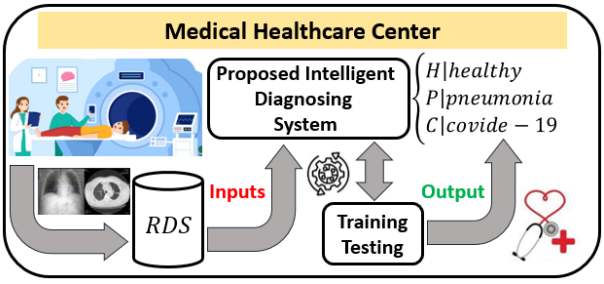


Fig. 6. Framework of the proposed system.

B. System Architecture

The objective of the proposed system is to, via learn mapping from inputs, predict/classify medical images to the correct class label. The proposed system is managed by six components. From a DL-based perspective, the components are grouped according to three phases, as shown in Fig. 7.

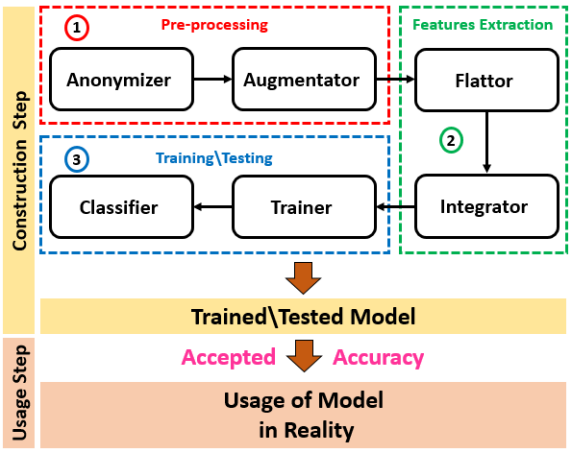


Fig. 7. Architecture of the proposed system.

Table I summarizes the main task of each component, the technique used for each task, and the quality attribute each task maintains.

TABLE I. Components

Name of component	Main task	Used technique	Quality attribute
Anonymizer	Privacy preservation	Pseudonym	Privacy
Augmentator	Augmentation	Cutmix\Mixup	Contrast
Patch Embedder	Solving positional information losses	PAE	Resistance
Integrator	Effective feature extraction	IbVTS	Performance
Trainer	Training on integrated extracted features	Distillation	Accuracy
Classifier	Prediction of output	Softmax	

C. Role of the Anonymizer Component

This component is responsible for protecting the privacy of patients by executing the pseudonym-based anonymity approach (PAA). For this, a pseudonym technique is employed. This technique is widely used to protect location privacy in location-based services. The key idea of the pseudonym technique is to handle personal data so that no data can be attributed to a specific person or data subject without additional information [25]. In this study, PAA was performed through three main steps: removing, replacing, and generalising. Fig. 8 illustrates these steps with an example.

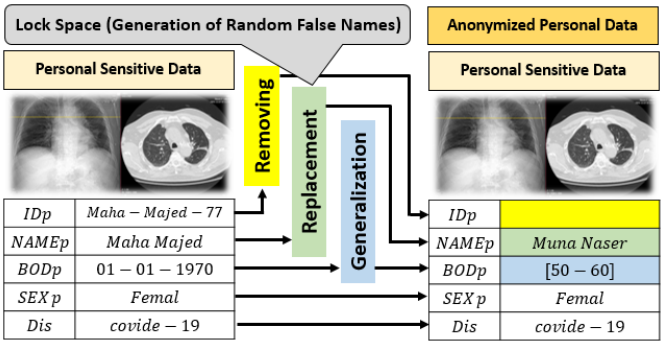


Fig. 8. Application of the three steps of the PAA.

As shown in Fig. 8, Maha Majed is the name of a real patient with a clear ID and clear date of birth, which, in turn, compromises this patient's privacy. Through using the PAA, their ID is removed. Their real name is replaced by "Muna Naser", which is a false name selected from the lock space that generates many pseudonyms based on some rules that help to create a false name adapted from the patient's real name. In our example, the real name consists of two parts (first name and last name), and so does the false one. In addition, the length of each part is 4 and 5 letters, respectively, which also matches the length of the first and last names of the pseudonym (false name). It is worth mentioning that privacy protection is a separate research field, and we will not discuss privacy protection in detail in this work, as we are only concerned about DL. The date of birth of the real patient is generalized in the anonymized profile so that no specific value is given. As for the sex and description fields, they are kept untouched. This way, the privacy of the patient is ensured, since the attacker cannot construct a malicious profile of a patient, and they also cannot make any inferences or link any information to a specific patient.

After performing the PAA task using the Anonymizer component, the resultant medical image ($\overline{\mathrm{MI}^{\mathrm{i}}_{\mathrm{type}\in(\mathrm{xr,ct})}}$) is privacy protected [see Eq. (3)].

$$(\overline{MI^{i}_{type\in(xr,ct)}}) = \langle \overline{NAMEp}, \overline{[BODp]}, SEXp, Dis \rangle | i = 1,2,\dots,Pat \quad (3)$$

D. Role of the Augmentator Component

This component is responsible for the augmentation of medical images. The objective of augmentation is to enhance the quality of training by enabling the extraction of more meaningful information. There is a wide spectrum of techniques used for augmentation [26]. The simplest techniques rely on some geometric transformation, such as rotating and cropping. However, such geometric transformation-based techniques are costly and complex, and they also require some manual operations to determine the angles and measures that suit the situation. Due to these limitations, researchers have moved towards more advanced techniques. Random erasing-based techniques such as the Cutout technique are used in this context [27]. The key idea behind the Cutout technique is to randomly remove/mask a square region of pixels in order to force the model to focus on a specific region and thus extract more robust features. For example, if the removed pixels form the background of a given image, then this noise will disappear. However, a critical weakness of the Cutout technique is related to its randomizing process. This means that sometimes it removes some informative parts of the given image, and consequently, some important features will be completely missing from the extracted space. The Mixup [28] technique saves all parts of a given image. The key idea behind this technique is to mix a pair of input images with their classes during training. Thus, more generalizable features that cover all input images are achieved. However, it may generate blurry images. An effective solution to the issue of blurring is discrimination, which is achieved by the use of the Cutmix technique [29]. Instead of masking the region, as is the case in the Cutout technique, it is replaced by the corresponding region from a different medical image. In this study, since we dealt with both X-ray- and CT scan-derived medical images, the Cutmix technique was used for the augmentation of the X-ray medical images, while the Mixup technique was used for the augmentation of the CT scan images. The reason behind this is that the ribs may blur X-ray scans, whereas there is no source of blurring in CT scans.

From a mathematical modeling perspective, let $\langle \overline{\alpha_A}, \overline{\beta_A} \rangle$ and $\langle \overline{\alpha_B}, \overline{\beta_B} \rangle$ represent two privacy-protected training samples, where $\overline{\alpha} = (\overline{\mathrm{MI}^{\mathrm{i}}_{\mathrm{type}\in(\mathrm{xr,ct})}})$ is a training image and $\overline{\beta} = \mathrm{Dis}$ is the corresponding label. For Mixup, the augmented sample is denoted as $(\overline{\overline{\alpha}}, \overline{\overline{\beta}})$, and the mixing process is given by the following Eq. (4) and Eq. (5):

$$\overline{\overline{\alpha}} = \delta\, \overline{\alpha_A} + (1-\delta)\, \overline{\alpha_B} \quad (4)$$

$$\overline{\overline{\beta}} = \delta\, \overline{\beta_A} + (1-\delta)\, \overline{\beta_B} \quad (5)$$

where, $\delta \in [0,1]$ is sampled from a beta distribution.

As for Cutmix, the generated augmented privacy-protected image is modeled as follows [see Eq. (6) and Eq. (7)]:

$$\overline{\overline{\alpha}} = \vartheta\, [\![\psi]\!]\, \overline{\alpha_A} + (1-\vartheta)\, [\![\psi]\!]\, \overline{\alpha_B} \quad (6)$$

$$\overline{\overline{\beta}} = \delta\, \overline{\beta_A} + (1-\delta)\, \overline{\beta_B} \quad (7)$$

where, $\vartheta \in \{0,1\}$ is a binary mask for determining which pixel is removed and replaced by the second image, and $[\![\psi]\!]$ is the multiplication operation.

From a visual perspective, the application of both Mixup and Cutmix to medical images is illustrated in Fig. 9 and Fig. 10, respectively.

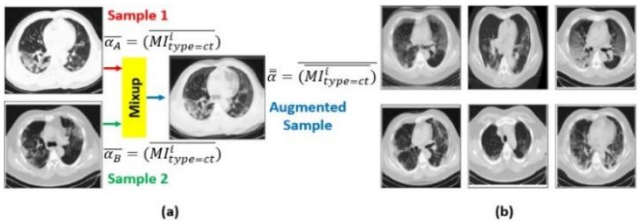


Fig. 9. Mixup-based augmentation. (a) Detailed operation; (b) different augmented samples.

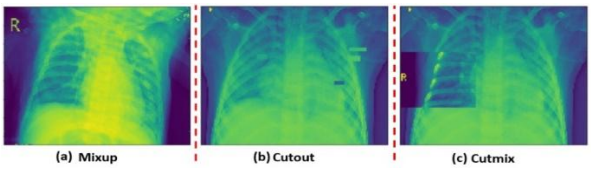


Fig. 10. Effectiveness of the Cutmix technique compared to Mixup and Cutout.

In Fig. 10(a), compared to Fig. 10(c), there are clear blurring regions that minimize the contribution of extracted features in the enhancement of training. Fig. 10(b) compared to Fig. 10(c), there are some informative parts out of the extracted space (the lower-right part of the left lung). In Fig. 10(c), the Cutmix technique avoids the pitfalls of both Mixup and Cutout, as a clear view of each specified part is used for the effective extraction of features.

E. Role of the Patch Embedder Component

This component is responsible for solving the problem of positional information losses by carrying out the Position-Aware Embedding (PAE) method on the augmented medical images. In detail, for a given augmented medical image ($\overline{\overline{\alpha}} = \overline{\overline{MI^{l}_{type\in(xr,ct)}}}$), which is generated from the source image ($MI^{l}_{type\in(xr,ct)}$), these two images are divided into parts of the same area, and then they are flattened into a sequence of non-overlapping patches. Then, corresponding positional information and learnable class tokens are manipulated, as shown in Fig. 11.

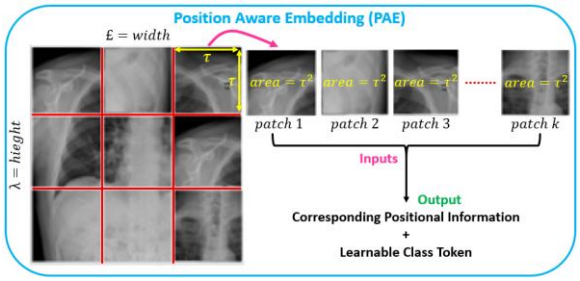


Fig. 11. Task of the patch embedder component when dealing with an X-ray image.

As shown in Fig. 11, λ and £ are the two dimensions of an augmented and privacy-protected medical image $\overline{\overline{\alpha}}$. The area of $\overline{\overline{\alpha}}$ is given by the following Eq. (8):

$$\overline{\overline{\alpha}}_{r} = \lambda \times £ \tag{8}$$

$\overline{\overline{\alpha}}$ is divided into a number of patches (each one has an area equal to $\tau \times \tau = \tau^2$). Relying on the total area of the medical image and the area of a patch, the number of patches k is given by the following Eq. (9):

$$k = \frac{\overline{\overline{\alpha}}_{r}}{\tau^2} \tag{9}$$

In terms of k patches, $\overline{\overline{\alpha}}$ is modeled as follows [see Eq. (10)]:

$$\overline{\overline{\alpha}} = \bigcup_{i=1}^{k} (\overline{\overline{\alpha}}^{\,i}_{pt}) \tag{10}$$

The generated flatted patches are passed to a linear embedding layer (LYR^{l}_{e}) to adjust their dimensions in a way that suits the dimensions of the model (DIM_{m}). The PAE method is applied to both the original image and the augmented instance. Fig. 12 provides a comprehensive view of a scene after the PAE method has been carried out.

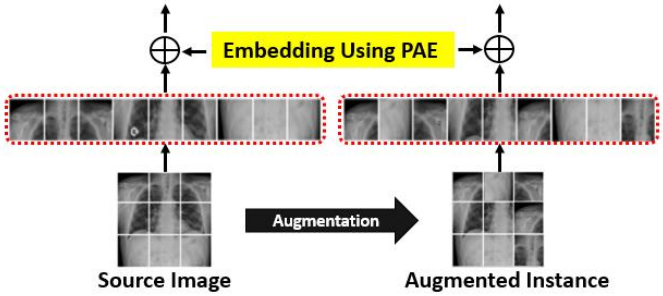


Fig. 12. Application of PAE on both a source image and an augmented image.

The application of PAE results in a series of patches that have the same spatial arrangement as in the source image. Let ($\mathfrak{T}_{lct}$) denote learnable class tokens and (PS_{info}) denote the positional information. Then, the output of the PAE is given by the following Eq. (11):

$PAE^{out}_{0} = \left[\mathfrak{T}_{lct}; \overline{\overline{\alpha}} LYR^{l}_{e}\right] + PS_{info}$, which can be detailed as follows:

$$PAE^{out}_{0} = \left[\mathfrak{T}_{lct}; \overline{\overline{\alpha}}^{\,1}_{pt} LYR^{l}_{e}; \overline{\overline{\alpha}}^{\,2}_{pt} LYR^{l}_{e}; \ldots; \overline{\overline{\alpha}}^{\,k}_{pt} LYR^{l}_{e}\right] + PS_{info} \tag{11}$$

where, $LYR^{l}_{e} \in \mathcal{R}^{(\tau^2 ch)\times DIM_m}$, and $PS_{info} \in \mathcal{R}^{(k+2)\times DIM_m}$.

F. Role of the Integrator Component

This component is responsible for ensuring effective feature extraction. It employs the concepts of the Vision Transformer (VisTrs) to carry out its mission [30]. The key idea behind the VisTrs is based on two fundamentals of performing image classification: 1) dealing with medical images as a series of patches that can be mapped into a semantic label, and 2) using a self-attention technique that enables effective feature extraction by handling the patches within an integration-based scan (i.e., integrate information at the level of the whole area of the image). This study adapts these two fundamentals and proposes an Integration-based Vision Transformer Structure (IbVTS). The structure of the VisTrs follows the encoder–decoder approach. In the context of the adaptation proposed in this work, the first fundamental of the VisTrs is satisfied by using a distill classifier (along with a corresponding distillation token), which will be discussed further on in this section when providing details on the role of the Trainer component. The second fundamental of the

VisTrs focuses on feature extraction and is satisfied as described below.

From a detail-based hierarchical perspective, the encoder of the VisTrs consists of a number of layers (NoL), and each layer contains two major units. The first unit is called multi-head self-attention (MhSa), which is responsible for carrying out the self-attention technique to ascertain the dependencies between the different patches that represent a given input image. The second unit is called the feed-forward unit (FeFoU) , which is responsible for generating a fully/densely connected network of features. Relying on a normalization layer ($\mathrm{LY_{nrm}}$), the MhSa and FeFoU are connected. From a mathematical perspective, the two units can be modeled as follows [see Eq. (12) and Eq. (13)]:

$$\widetilde{\mathrm{PAE}_{\varphi}^{\mathrm{out}}} = \mathrm{MhSa}\left\langle \mathrm{LY_{nrm}}\left(\mathrm{PAE}_{\varphi-1}^{\mathrm{out}}\right)\right\rangle + \mathrm{PAE}_{\varphi-1}^{\mathrm{out}} | \varphi = 1,2,\ldots,\mathrm{NoL} \quad (12)$$

$$\mathrm{PAE}_{\varphi}^{\mathrm{out}} = \mathrm{FeFoU}\left\langle \mathrm{LY_{nrm}}\left(\widetilde{\mathrm{PAE}_{\varphi}^{\mathrm{out}}}\right)\right\rangle + \widetilde{\mathrm{PAE}_{\varphi}^{\mathrm{out}}} | \varphi = 1,2,\ldots,\mathrm{NoL} \quad (13)$$

The operations that are executed within the self-attention technique are as follows:

1) For a given input sequence of image features, $\begin{cases} k: key \\ q: query \\ v: value \end{cases}$ are calculated for each element included in the sequence.

2) Relying on the inner/dot product, the relevance between the current element and other elements is obtained. In other words, this step highlights and integrates the relative importance of the patches in the sequence.

3) The results of step 2 are scaled based on the dimension of key ($\mathrm{K_{dim}}$) as a scaling factor and then used as inputs for the Softmax function.

4) To focus on more important values, the output of the Softmax function is multiplied by v.

Eq. (14) and Eq. (15) summarize the operations of the self-attention technique.

$$\begin{cases} \mathrm{k: key} \\ \mathrm{q: query} \\ \mathrm{v: value} \end{cases} = \mathrm{PAE^{input}}\ \mathbb{U}_{\mathrm{k,q,v}} \left| \mathbb{U}_{\mathrm{k,q,v}} \in \mathcal{R}^{\mathrm{DIM_m} \times 3\mathrm{K_{dim}}} \right. \quad (14)$$

$$\mathrm{SeAt}(\mathrm{PAE^{input}}) = \mathrm{softmax}\left(\mathrm{qk^T}/\sqrt{\mathrm{K_{dim}}}\right).\mathrm{v} \quad (15)$$

The environment (patches) within which the self-attention (SeAt) is executed allows for the parallel execution of these operations. Here, the MhSa, considered an extension of SeAt, comes to the fore. To activate self-attention (SeAt) in a parallel manner, instead of using a single value for the triplet (k, q, v), multiple values are used. Consequently, we deal with (h) heads (i.e., a head for each attention). The results of all the attention heads are concatenated to express the first unit of the encoder of the VisTrs [see Eq. (16)].

$$\mathrm{MhSa}\left(\mathrm{PAE^{input}}\right) = \mathrm{concat}\left\langle \mathrm{SeAt_1}(\mathrm{PAE^{input}}); \mathrm{SeAt_2}(\mathrm{PAE^{input}}); \ldots; \mathrm{SeAt_h}(\mathrm{PAE^{input}}); \right\rangle \langle \mathrm{lw} \rangle \quad (16)$$

where, lw represents the learnable weights, $\mathrm{lw} \in \mathcal{R}^{\mathrm{h.K_{dim}} \times \mathrm{DIM_m}}$.

As for FeFoU, it is constructed by a fully connected layer, followed by GeLU, followed by another fully connected layer.

Fig. 13 presents the Integrator component, which is based on the Patch Embedder component.

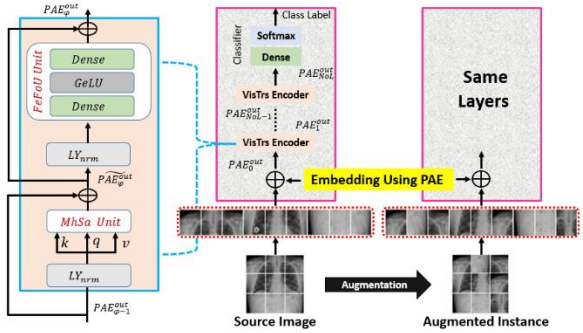


Fig. 13. A VisTrs architecture-based Integrator built based on the mission of the patch embedder component.

G. Role of the Trainer and Classifier Components

These components are responsible for training the model in a way that enhances both model performance and prediction accuracy. Originally, the VisTrs is driven from the data-efficient image transformer [31], where minimizing training is a major goal based on the student–teacher approach. The student–teacher approach relies on a distillation token generated from the augmented image instance. So, the left part of Fig. 13 represents the student, and the right part represents the teacher. Based on this, Eq. (11) is updated to become the following Eq. (17):

$$\mathrm{PAE_0^{out}} = \left[\mathfrak{T}_{\mathrm{lct}}; \mathfrak{T}_{\mathrm{dst}}; \overline{\overline{\alpha}}_{\mathrm{pt}}^{1}\mathrm{LYR_e^l}; \overline{\overline{\alpha}}_{\mathrm{pt}}^{2}\mathrm{LYR_e^l}; \ldots; \overline{\overline{\alpha}}_{\mathrm{pt}}^{\mathrm{k}}\mathrm{LYR_e^l}\right] + \mathrm{PS_{info}} \quad (17)$$

where, $\mathrm{LYR_e^l} \in \mathcal{R}^{(\tau^2 \mathrm{ch}) \times \mathrm{DIM_m}}$, $\mathrm{PS_{info}} \in \mathcal{R}^{(\mathrm{k}+2) \times \mathrm{DIM_m}}$, and $\mathfrak{T}_{\mathrm{dst}}$ denote the distillation token. In other words, the Trainer uses extracted features to train two classifiers, where each one has a dense layer followed by the Softmax function. The student classifier provides the class token, while the teacher classifier provides the distill token. Both of them are used as inputs for the fusion layer, which provides the final class of the medical image according to the following Eq. (18):

$$\mathrm{C_{final}} = \frac{1}{2} \times \left(\mathrm{C_{token}^{class}} + \mathrm{C_{token}^{distill}}\right) \quad (18)$$

Fig. 14 illustrates the Trainer's task, which is based on that of the Integrator.

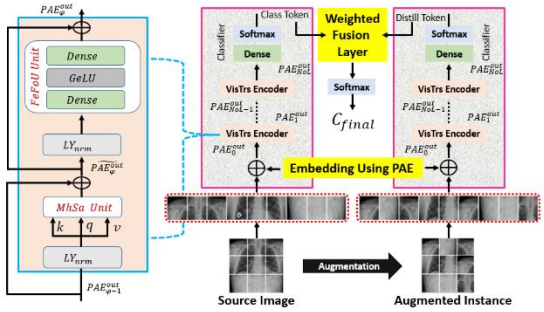


Fig. 14. Mission of the trainer component.

In reality, in the context of classification, a given medical image can be classified as healthy/normal, COVID-19, or pneumonia. To enable binary (healthy or COVID-19) or multi-class modes (binary plus pneumonia), a loss function is used in

the Classifier component, and this function is expressed as follows [see Eq. (19)]:

$$\mathcal{Q}\left(\mathfrak{V}_{ij}, \mathfrak{W}_{ij}\right) = \frac{-1}{pat}\sum_{i=1}^{trg}\sum_{j=1}^{ds} \mathfrak{W}_{ij} \log\left(\frac{1}{1+e^{-\mathfrak{V}_{ij}}}\right) + \left(1 - \mathfrak{W}_{ij}\right) \log\left(1 - \frac{1}{1+e^{-\mathfrak{V}_{ij}}}\right) \quad (19)$$

where, trg, ds are the number of training images and defined classes. $\mathfrak{W}_{ij}$ represents the ground-truth labels. $\mathfrak{V}_{ij}$ is the predicted probability.

H. Details of the Proposed System's Architecture

A sequence diagram, Fig. 15, is used to show the tasks of the components involved in the architecture of the proposed system.

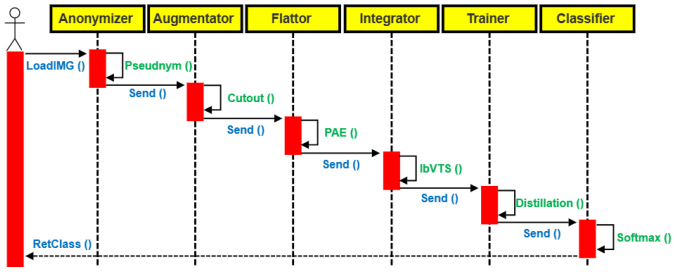


Fig. 15. Sequence diagram based on architecture details.

IV. Experimental Results and Evaluation

In this work, experiments were conducted on two different datasets to construct (train) and test the proposed diagnosing system. One of these datasets is the COVIDx dataset, created to store X-ray medical images [32]. This dataset consists of 13,962 X-ray images distributed over three classes, as summarized in Table II. We also used the SARS-CoV-2 dataset, which contains CT scan-derived medical images [33]. This dataset comprises 2,482 CT scan images distributed over two classes, as summarized in Table III.

TABLE II. Description of the COVIDx Dataset

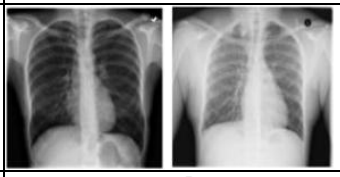
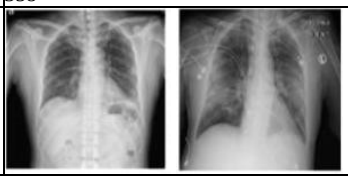
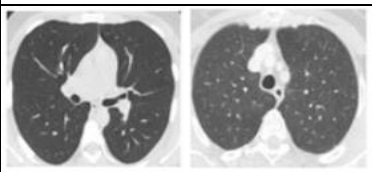
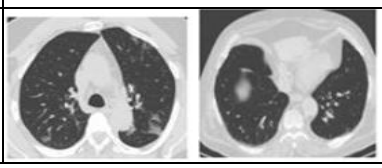
Class 1	Healthy
Number of images	8066
Samples	
Class 2	Pneumonia
Number of images	5538
Samples	
Class 3	COVID-19
Number of images	358
Samples	

TABLE III. Description of the COVIDx Dataset over two classes

Class 1	Healthy
Number of images	1230
Samples	
Class 2	COVID-19
Number of images	1252
Samples	

A. Types of Evaluation

As shown in Fig. 16, we used two types of evaluation metrics: visual evaluation and numeric evaluation.

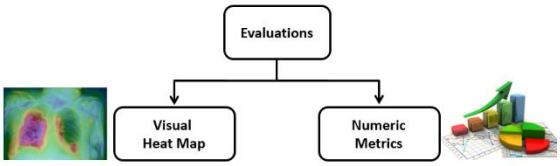


Fig. 16. Types of evaluation metrics used in the present study.

B. Used Metrics

Four metrics were used in this study for evaluation purposes. All of them were derived from the confusion matrix summarized in Table IV.

TABLE IV. Confusion Matrix

Predicted class Actual class	C_{final}	$\neg C_{final}$	Sum
C_{final}	TP	FN	S
$\neg C_{final}$	FP	TN	Su
Sum	$\widetilde{S}$	$\widetilde{Su}$	All

The elements of the above confusion matrix are as follows:

- True positives (TPs): The positive images that were correctly labeled by the classifier.
- True negatives (TNs): The negative images that were correctly labeled by the classifier.
- False positives (FPs): The negative images that were incorrectly labeled as positive (e.g., images of class healthy = no for which the classifier predicted healthy = yes).
- False negatives (FNs): The positive images that were mislabeled as negative (e.g., images of class healthy = yes for which the classifier predicted healthy = no).

$$\text{Accuracy} = \frac{TP+TN}{All} \quad (20)$$

Accuracy refers to the recognition rate, which quantifies the percentage of test set images that have been correctly classified.

In other words, it reflects how well the classifier recognizes images from the various classes [see Eq. (20)].

$$\text{Precision} = \frac{\text{TP}}{\text{S}} \quad (21)$$

Precision refers to the exactness, which refers to the percentage (%) of images that the classifier correctly labeled as positive. Therefore, it determines how many of the positive predictions are correct [see Eq. (21)].

$$\text{Recall} = \frac{\text{TP}}{\text{S}} \quad (22)$$

Recall (also known as sensitivity) refers to the completeness, which refers to the percentage (%) of positive tuples that the classifier labeled as positive. Recall is used as a measure of a classifier's tendency to identify infected cases [see Eq. (22)].

$$\text{Specificity} = \frac{\text{TN}}{\text{Su}} \quad (23)$$

Specificity refers to the true negative recognition rate. In other words, it measures the ability of a classifier to detect non-infected cases [see Eq. (23)].

It is worth mentioning that in terms of semantic meaning, a high value for any of the metrics mentioned above is preferable and reflects better performance, and vice versa.

Regarding visual evaluation, heat maps were employed to reflect the attention maps used in the learning process. Heat maps are used to illustrate the gradually increasing focus on regions of interest over layers during the learning process. This, in turn, reflects how regions are tightly coupled with the predicted class when it comes to the visual monitoring of progress.

It is worth mentioning that experimrnts are conducted in the Artificial Intelligence and Multi Media (AIMD) Lab within the campus of Fahad Bis Sultan University (FBSU) at Computing College.

C. Evaluation Based on X-ray Dataset

The COVIDx dataset was divided into a training part (25 %) and a testing part (75 %), taking into account the four metrics used for each class. Table V summarizes the results obtained with and without using augmentation.

TABLE V. EXPERIMENTAL RESULTS BASED ON X-RAY IMAGES FROM THE COVID DATASET

Approach	Term Metric	AVG	Class		
			Healthy	***COVID-19***	***Pneumonia***
Without Augmentation	Accuracy	≈ 87.16	88.23	87.77	85.47
	Precision	≈ 87.37	87.11	86.81	88.2
	Recall	≈ 87.84	89.21	88	86.31
	Specificity	≈ 87.11	86.26	87.62	87.45
With Augmentation	Accuracy	≈ 92.62	95.88	90	91.98
	Precision	≈ 93.47	94.63	92.12	93.66
	Recall	≈ 93.22	97.52	90	92.13
	Specificity	≈ 95.98	93.83	98.57	95.53

The values documented in Table V represent good performance with regard to diagnosing positive and negative cases. However, to highlight the effectiveness of the proposed augmentation technique, Fig. 17 compares the averages of the two approaches (with and without augmentation) for each class.

Fig. 17 reflects considerable enhancements for the four metrics when using augmentation, where it contributes 5.46, 6.1, 5.38, and 8.87 enhancements for accuracy, precision, recall, and specificity, respectively. The reason behind this is related to the effective extraction of features used for training, since the augmentation technique used enabled an increased focus on the regions of interest. This justification is visually supported by the heat map shown in Fig. 18.

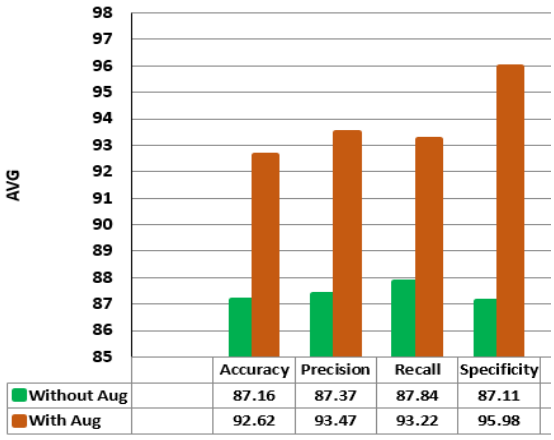


Fig. 17. Comparison between the augmentation-based approach and the non-augmentation-based approach on the X-ray dataset.

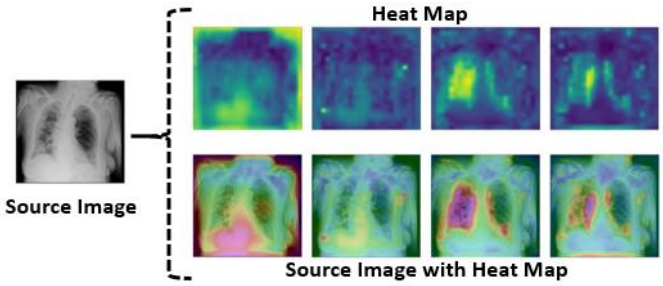
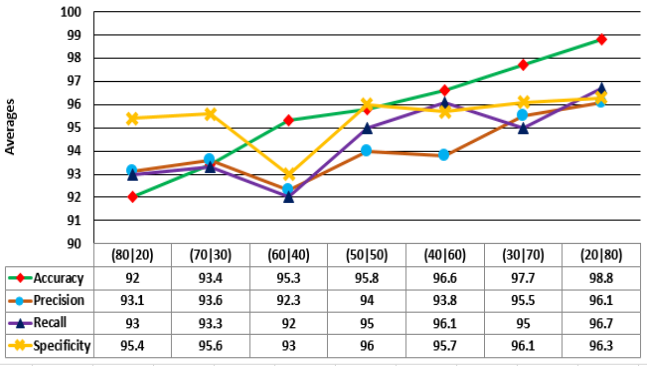


Fig. 18. Heat map of a source COVID-19 X-ray image.

Fig. 18 shows the gradually increasing focus on the zones that are informative for diagnosing/predicting illness. The heat map above reflects the confused focus on random zones of the lung and the point at which the focus becomes more specific on the zones that are tightly coupled with the class of the image, reaching the zones that represent COVID-19 in terms of medical diagnosis.

To measure the resistance of the proposed system against a small-sized training dataset (and to, in turn, evaluate its potential for usage in reality), we repeated our experiments using a different training/testing data ratio (stepwise increase of 10 %), meaning that 20 % of the COVIDx dataset was used for training and 80 % was used for testing), as shown in Fig. 19.



	(80|20)	(70|30)	(60|40)	(50|50)	(40|60)	(30|70)	(20|80)
Accuracy	92	93.4	95.3	95.8	96.6	97.7	98.8
Precision	93.1	93.6	92.3	94	93.8	95.5	96.1
Recall	93	93.3	92	95	96.1	95	96.7
Specificity	95.4	95.6	93	96	95.7	96.1	96.3

Fig. 19. Average of four metrics over different training/testing data splits.

Fig. 19 presents the results of seven experiments, each of which had its own unique training/testing division ratio. For example, in our second experiment, 30% of the dataset was allocated for training, and 70% was allocated for testing, meaning that this split can be expressed as 30|70. Regarding accuracy, it is obvious that there was a gradual increase in enhancement from about 92% in the first experiment to about 99% in the seventh one. This means that allocating more data for training leads to greater accuracy. However, there is a gap (about 8.8) in accuracy values between the 20|80 training ratio and its counterpart, the 80|20 training ratio. However, in both cases, the accuracy is still within a considerable range, which is a good indicator of the proposed system's potential for use in reality. The other metrics fluctuated randomly in response to the different training ratios. In this context, it is worth mentioning that accuracy is the most important measure to consider when evaluating a system.

D. Evaluation Based on CT Scan Dataset

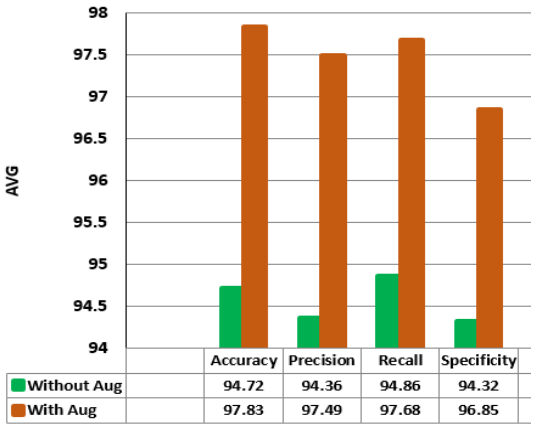
To facilitate a fair comparison between our evaluation of the proposed system based on X-ray images and CT scan images, the same evaluation procedure followed for our experiments on the X-ray images was repeated. Therefore, the SARS-CoV-2 dataset was divided into a training part (25 %) and a testing part (75 %), taking into account the four metrics used for each class. Table VI summarizes the results obtained with and without using augmentation.

TABLE VI. EXPERIMENTAL RESULTS BASED ON CT SCAN IMAGES FROM THE SARS-COV-2 DATASET

Approach	Term Metric	AVG	Class	
			Healthy	*COVID-19*
Without Augmentation	Accuracy	≈ 94.72	94.44	95
	Precision	≈ 94.36	93.88	94.83
	Recall	≈ 94.86	94.69	94.98
	Specificity	≈ 94.32	94.54	94.09
With Augmentation	Accuracy	≈ 97.83	97.55	98.11
	Precision	≈ 97.49	97.21	97.76
	Recall	≈ 97.68	97.33	98.03
	Specificity	≈ 96.85	96.57	97.13

The values documented in Table VI indicate that the model's performance was better than that achieved via the use of the X-ray images in terms of diagnosing positive and negative cases. However, to highlight the effectiveness of the proposed augmentation technique, Fig. 20 compares the averages of the two approaches (with and without augmentation) for each class.

Fig. 20 shows that considerable improvements in the four metrics were achieved when using augmentation, as enhancements of 3.11, 3.13, 2.82, and 2.53 were recorded for accuracy, precision, recall, and specificity, respectively. The results derived from the use of CT scan images are better than those derived from the use of X-ray images. The CT scan-derived results' superiority can be attributed to the ability of the Mixup augmentation technique to overcome the blurring issue that arises due to the ribs obscuring features in X-ray images. This justification is visually supported by the heat map shown in Fig. 21.



	Accuracy	Precision	Recall	Specificity
Without Aug	94.72	94.36	94.86	94.32
With Aug	97.83	97.49	97.68	96.85

Fig. 20. Comparison between the augmentation-based approach and the non-augmentation-based approach when using the CT-scan dataset.

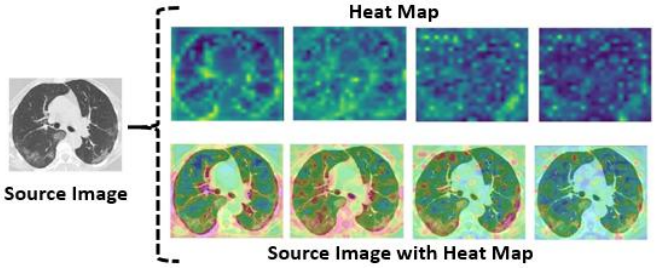
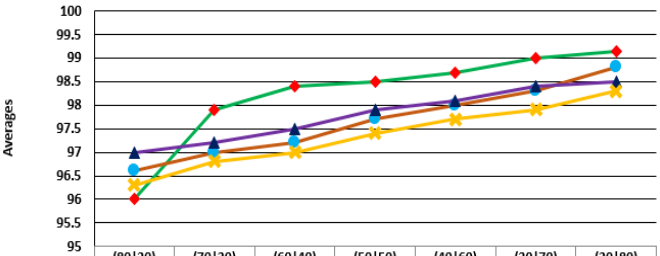


Fig. 21. Heat map of a source COVID-19 CT scan image.

As was the case with the X-ray images presented in Fig. 18, Fig. 21 shows the gradually increasing focus on the zones that are informative for diagnosing/predicting illness. In other words, the system learned to highlight sets of pixels that are strongly linked with illness. It is worth mentioning that the CT scanning technique is more accurate than X-ray imaging, which means that learning on CT scan images leads to higher levels of accuracy, especially if they are supported by augmentation, since augmentation leads to the extraction of more meaningful features.

To measure the resistance of the proposed system against a small-sized CT-based training dataset, the same procedure adopted to assess the resistance of the X-ray dataset was repeated, as shown in Fig. 22.

The results of the seven experiments (shown in Fig. 22) provide strong proof of the effectiveness of applying augmentation on CT scan images to increase the level of accuracy and other metrics as the training ratio increases. Steady enhancements for all metrics were recorded. Although the accuracy increases, there is a small gap (3.15) between the accuracy value when using an 80|20 testing and training ratio and that when using a 20|80 testing and training ratio. This proves that the proposed system has the ability to be trained on a small amount of data, which further indicates its potential for use in reality.



	(80|20)	(70|30)	(60|40)	(50|50)	(40|60)	(30|70)	(20|80)
Accuracy	96	97.9	98.4	98.5	98.7	99	99.15
Precision	96.6	97	97.2	97.7	98	98.3	98.8
Recall	97	97.2	97.5	97.9	98.08	98.4	98.5
Specificity	96.3	96.8	97	97.4	97.7	97.9	98.3

Fig. 22. Average of the four metrics over different training/testing data splits in the CT-based training dataset.

E. Security Analyzing

To analyze the proposed model's resistance against compromising the privacy of patients, let P^{e}_{Guess} denote the probability that a malicious party can successfully guess whether the event e is true. The proposed privacy protection method is resistant if [see Eq. (24)]:

$$P^{e_1=\text{personal info}_i \in RDS}_{Guess} = P^{e_1=\text{personal info}_j \in RDS}_{Guess} \forall\ (0 < i \neq j \leq pat) \quad (24)$$

Proof: The attacker cannot obtain any benefit from employing their side's information to recognize the patient. This is because the lock space prevents the attacker from knowing the ID of the patient. Moreover, the first name and last name are masked by dummies of the same size (i.e., a first and last name with the same number of letters as the patient's first and last name), which, in turn, leads to more confusion on the attacker's side. Furthermore, the generalization tactic applied to the date of birth of the patient does not reveal any specific information about the age of the patient. Only the attacker can know the gender of the patient and whether they have COVID-19. As a result, the attacker is forced to randomly guess the personal information of the patient, which leads to uncertainty in inferring sensitive information when attempting to compromise a patient's privacy and potentially harm them.

V. Conclusion

In this study, a deep learning-based system dedicated to detecting COVID-19 using X-ray- and CT scan-derived medical images is presented. The system is managed by six components. The Anonymizer is responsible for preserving the privacy of patients through the use of a pseudonym-based anonymity approach. The Augmentator is responsible for increasing the quality of the medical images through the use of augmentation techniques (Cutmix and Mixup for X-ray and CT scan images, respectively). The Patch Embedder is responsible for solving the problem of positional information loss by applying the Position-Aware Embedding method to the augmented medical images. The Integrator is responsible for effective feature extraction, which is carried out through employing the concepts of the Vision Transformer Structure. The Trainer and Classifier components are responsible for training the proposed system based on the student–teacher approach and generating the final classes via the Softmax function. The COVIDx (X-ray images) and SARS-CoV-2 (CT-scan images) datasets were utilized in this study to train and test the proposed system. Our evaluations, which were based on accuracy, precision, recall, and specificity metrics, showed that there was an enhancement in values when using augmentation. However, the enhancement achieved after training on CT scan images was better than that achieved after training on X-ray images. Moreover, the system was evaluated using a small amount of data by reversing the training/testing ratios. The results corresponding to the use of X-ray images were promising, as the system achieved 92% accuracy when 20% of the data were allocated for training and 80% of the data were allocated for testing and 98.8% accuracy when 80% of the data were allocated for training and 20% of the data were allocated for testing. However, compared to the results derived from the use of X-ray images, the results corresponding to the use of CT scan images were better, with an accuracy value of 96% being achieved after 20% of the data were allocated for training and 80% of the data were allocated for testing, as well as an accuracy value of 99.15% being achieved after 80% of the data were allocated for training and 20% of the data were allocated for testing. The key limitation of this work is related to the fact that we ignored evaluating performance in terms of the time taken to generate predictions and complexity. In future work, we will address this limitation by adopting a parallel training approach. In addition, testing the proposed system using additional datasets as well as including statistical significance testing (e.g., confidence intervals, p-values) will be in future work.

References

[1] Guo, Yan-Rong, et al. "The origin, transmission and clinical therapies on coronavirus disease 2019 (COVID-19) outbreak–an update on the status." Military Medical Research 7.1 (2020): 1-10.

[2] Lauer, Stephen A., et al. "The incubation period of coronavirus disease 2019 (COVID-19) from publicly reported confirmed cases: estimation and application." Annals of internal medicine 172.9 (2020): 577-582.

[3] Nawaz, Ahsan, et al. "Epidemic spread and its management through governance and leadership response influencing the arising challenges around COVID-19 in Pakistan—a lesson learnt for low income countries with limited resource." Frontiers in public health 8 (2020): 573431.

[4] Chen, Wei, et al. "Reducing false negatives in COVID-19 testing by using microneedle-based oropharyngeal swabs." Matter 3.5 (2020): 1589-1600.

[5] Iliescu, F. S., et al. "Point-of-Care Testing-the Key in the Battle against SARS-CoV-2 Pandemic. Micromachines 2021, 12, 1464." (2021).

[6] Koyyada, Shiva Prasad, and Thipendra P. Singh. "A Systematic Survey of Automatic Detection of Lung Diseases from Chest X-Ray Images: COVID-19, Pneumonia, and Tuberculosis." SN Computer Science 5.2 (2024): 229.

[7] Alafif, Tarik, et al. "Machine and deep learning towards COVID-19 diagnosis and treatment: survey, challenges, and future directions." International journal of environmental research and public health 18.3 (2021): 1117.

[8] Cossio, Manuel. "Augmenting Medical Imaging: A Comprehensive Catalogue of 65 Techniques for Enhanced Data Analysis." arXiv preprint arXiv:2303.01178 (2023).

[9] Abdollahi, Behnaz, Naofumi Tomita, and Saeed Hassanpour. "Data augmentation in training deep learning models for medical image analysis." Deep learners and deep learner descriptors for medical applications (2020): 167-180.

[10] Altaf, Fouzia, et al. "Going deep in medical image analysis: concepts, methods, challenges, and future directions." IEEE Access 7 (2019): 99540-99572.

[11] Alafeef, Maha, and Dipanjan Pan. "Diagnostic approaches for COVID-19: lessons learned and the path forward." ACS nano 16.8 (2022): 11545-11576.

[12] Wu, Song, et al. "Deep Learning and Medical Imaging for COVID-19 Diagnosis: A Comprehensive Survey." arXiv preprint arXiv:2302.06611 (2023).

[13] Panday, Aishwarza, Muhammad Ashad Kabir, and Nihad Karim Chowdhury. "A survey of machine learning techniques for detecting and diagnosing COVID-19 from imaging." arXiv preprint arXiv:2108.04344 (2021).

[14] Awassa, Lamia, et al. "Study of Different Deep Learning Methods for Coronavirus (COVID-19) Pandemic: Taxonomy, Survey and Insights." Sensors 22.5 (2022): 1890.

[15] Wang, Xinggang, et al. "A weakly-supervised framework for COVID-19 classification and lesion localization from chest CT." IEEE transactions on medical imaging 39.8 (2020): 2615-2625.

[16] Singh, Shrinjal, et al. "CNN based Covid-aid: Covid 19 Detection using Chest X-ray." 2021 5th International Conference on Computing Methodologies and Communication (ICCMC). IEEE, 2021

[17] Lu, Siyuan, et al. "An explainable framework for diagnosis of COVID-19 pneumonia via transfer learning and discriminant correlation analysis." ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM) 17.3s (2021): 1-16.

[18] Sahinbas, Kevser, and Ferhat Ozgur Catak. "Transfer learning-based convolutional neural network for COVID-19 detection with X-ray images." Data science for COVID-19. Academic Press, 2021. 451-466.

[19] Chaudhary, Suman, and Yan Qiang. "Ensemble deep learning method for Covid-19 detection via chest X-rays." 2021 Ethics and Explainability for Responsible Data Science (EE-RDS). IEEE, 2021.

[20] Upadhyay, Kamini, Monika Agrawal, and Desh Deepak. "Ensemble learning - based COVID - 19 detection by feature boosting in chest X - ray images." IET Image Processing 14.16 (2020): 4059-4066.

[21] Goel, Tripti, et al. "Automatic screening of COVID-19 using an optimized generative adversarial network." Cognitive computation (2021): 1-16.

[22] Ardakani, Ali Abbasian, et al. "Application of deep learning technique to manage COVID-19 in routine clinical practice using CT images: Results of 10 convolutional neural networks." Computers in biology and medicine 121 (2020): 103795.

[23] Er, Mehmet Bilal. "COVID - 19 detection based on pre - trained deep networks and LSTM model using X - ray images enhanced contrast with artificial bee colony algorithm." Expert Systems 40.3 (2023): e13185.

[24] Hasan, Ali M., et al. "Classification of Covid-19 coronavirus, pneumonia and healthy lungs in CT scans using Q-deformed entropy and deep learning features." Entropy 22.5 (2020): 517.

[25] Tyagi, Amit Kumar, and N. Sreenath. "A comparative study on privacy preserving techniques for location based services." British Journal of Mathematics & Computer Science 10.4 (2015): 1-25.

[26] Chlap, Phillip, et al. "A review of medical image data augmentation techniques for deep learning applications." Journal of Medical Imaging and Radiation Oncology 65.5 (2021): 545-563.

[27] DeVries, Terrance, and Graham W. Taylor. "Improved regularization of convolutional neural networks with cutout." arXiv preprint arXiv:1708.04552 (2017).

[28] Zhang, Hongyi, et al. "mixup: Beyond empirical risk minimization." arXiv preprint arXiv:1710.09412 (2017).

[29] Yun, Sangdoo, et al. "Cutmix: Regularization strategy to train strong classifiers with localizable features." Proceedings of the IEEE/CVF international conference on computer vision. 2019.

[30] Yang, Jianwei, et al. "Focal self-attention for local-global interactions in vision transformers." arXiv preprint arXiv:2107.00641 (2021).

[31] Touvron, Hugo, et al. "Training data-efficient image transformers & distillation through attention." International conference on machine learning. PMLR, 2021.

[32] Wang, Linda, Zhong Qiu Lin, and Alexander Wong. "Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images." Scientific reports 10.1 (2020): 19549.

[33] Soares, Eduardo, et al. "SARS-CoV-2 CT-scan dataset: A large dataset of real patients CT scans for SARS-CoV-2 identification." MedRxiv (2020): 2020-04.

Navigating the Landscape of Automated Information Extraction for Financial Fund Prospectuses: Survey and Challenges

Yuyao Xu, Mohamad Farhan Mohamad Mohsin
School of Computing (SoC), University Utara Malaysia (UUM), 06010 Sintok, Kedah, Malaysia

***Abstract*—In the financial sector, a fund prospectus is a critical document mandated by the Securities and Exchange Commission (SEC) that provides vital information about investments to the public. These documents encompass a range of financial concepts that define the fund's operations, including its name and disclaimers associated with periodic reports. Traditionally, the identification of these concepts has been a manual, labour-intensive, and costly task for financial regulators, aimed at ensuring the completeness of information. Automating this process is fraught with challenges, including the lengthy nature of prospectuses, the nuances of financial language, and the scarcity of labelled data for effective model training. This study explores state-of-the-art methods for information extraction, specifically within the context of financial documents. It begins with an overview of information extraction, detailing its definition and various types, such as Named Entity Recognition (NER) and event extraction. The discussion highlights the increasing significance of information extraction in the financial domain and reviews typical application areas. Ultimately, this research seeks to highlight the challenges within existing methods through a comprehensive literature review, emphasizing the need for more effective techniques tailored to the extraction of financial concepts in fund prospectuses. By enhancing and streamlining the extraction process, it aspires to improve efficiency and reduce costs for financial regulators, thereby ensuring more accurate and comprehensive information dissemination.**

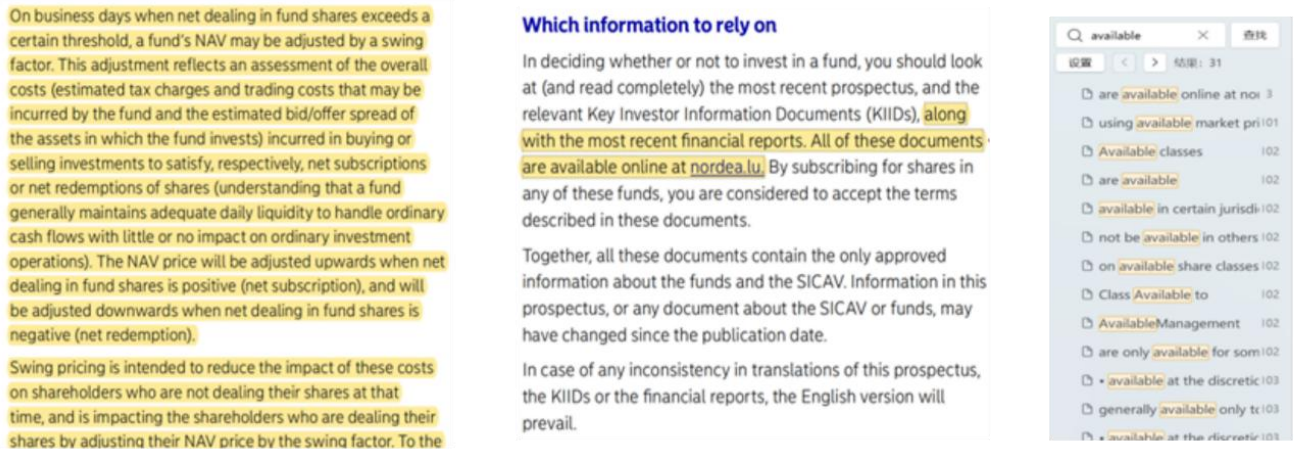
Keywords*—Machine learning; information automation; financial documentation

I. Introduction

Financial documents (also known as financial statements) are formal records that provide a comprehensive and accurate representation of an enterprise's financial status, performance, and cash flow over a specific period. There are many types of financial documents, such as financial news, discussion boards, annual financial reports, and fund prospectuses. In the financial sector, every investment fund is mandated to provide detailed and informative financial documentation. These documents enable regulatory authorities and investors to comprehensively evaluate all pertinent and material information regarding the investment. For example, the prospectus offers an in-depth account of the public investment opportunity. Such documentation, primarily conveyed through natural language, supplemented with tables and mathematical formulations, serves as critical instruments in safeguarding regulatory compliance and ensuring transparency and oversight of the investment. These financial documents are highly regulated, which are important to every company. The regulations that apply depend on the type of fund to which the documents refer and on the countries in which the fund is distributed. Specifically, the regulations clearly outline the specific information that must be included in each type of document. Based on these requirements, legal experts can develop 'suggested wording' to ensure that fund documents are both compliant with regulatory standards and accurately convey the necessary details. The suggested wording is by no means universal, since asset and fund managers customize the narrative of fund documents to differentiate their products from those of the competitors. Furthermore, both funds and regulations change over time; as a consequence, it is important to update the corresponding fund documentation accordingly.

When financial documents are submitted to the financial regulator, the regulator needs to manually read the financial document and check the correctness of it. One of the most information factors that affects the correctness is that the financial document should clearly specify all the information needed in each type of document. The current practice for finding the sentences related to each type of information in the financial documents is still largely a manual activity [1, 2]. The financial regulator has to search for some “keywords” for each type of information, and check if the returned results are complete. This searching process is time-consuming and costly [1]. On one hand, there could be hundreds of types of information. The regulator has to manually type keywords related to all these types of information to completely check the financial documents, which is time-consuming. Besides that, many sentences can match the same keyword [3]. The regulator must select the right sentences describing each type of information to understand whether the sentences correctly explain the corresponding type of information. This process is costly. Doing so often turns out to be a complex, time-consuming and expensive activity, noting that the relevant information may not be in a structured form but rather provided through textual statements. For example, Fig. 1 shows two snippets of financial documents (i.e., prospectuses). The two snippets show a part of text churn related to the financial concepts “swing pricing” and “disclaimer on periodical reports". The text related to each financial concept is quite different. For swing pricing, a very long text churn is labelled as related to it [see Fig. 1(a)]; while there is only one sentence related to ‘disclaimer on periodical reports’ [see Fig. 1(b)], which shows that the report is available online. Such text churn is not easy to locate. If the regulator searches some keywords

(e.g., available), 31 results are returned, as in Fig. 1(c). The regulator has to examine 31 paragraphs to find the text related to 'disclaimer on periodical reports. An alternative is to locate the text based on the title of paragraph; however, there are still hundreds of titles in the document. The regulator may not easily know that the text related to 'disclaimer on periodical reports' is under the paragraph with the title "which information to rely on".



(a) swing pricing (b) Disclaimer on periodical reports (c) Result of keyword search

Fig. 1. Examples of financial documents and financial concepts.

II. Motivation

A fund prospectus is a formal legal document required by regulatory authorities, such as the Securities and Exchange Commission (SEC), that provides detailed information about an investment fund [4]. It is designed to inform potential investors about the fund's objectives, strategies, risks, fees, performance, and other essential details to help them make informed investment decisions. The significant manual effort required frequently results in increased fund setup costs, extended time-to-market for fund products, and heightened compliance risks. There is therefore a need for automating finding the sentences related to each type of information. By doing so, the financial regulator can directly find the necessary sentences without manual checking. If the automated identification is accurate, the regulator also does not need to select the right sentences from all the returned results. In this study, the type of information to be extracted is referred to as financial concepts, which explain the concept, behaviour, definition, and metadata of financial activities, such as the detailed explanation of the calculation methodology for determining the issue price, along with the specific conditions and procedures governing the issuance process. All these concepts need to be clearly explained in certain types of financial documents.

Existing studies focus on mining financial data from web media (e.g., financial news and discussion boards [5]) and traditional financial documents (e.g., annual financial reports and 10-K) [6]. Besides, tracing information in documents has been widely studied in the areas of information extraction and software engineering [7, 8]. In these areas, they extract named entities from news and block [9]; based on different entities, the events related to each entity are also extracted [10, 11]. However, all these methods are not designed for the need of financial concept extraction in financial documents, which are more complex documents compared to discussion boards in the financial domain.

This study is proposed by analysing financial concepts in financial documents, taking fund prospectus documents in stock markets as a case study, and developing a tool-supported solution that can analyse a financial document (i.e., fund prospectuses) to automatically extract key types of information (i.e., financial concepts). The selection of the fund prospectus for the case study is due to its special characteristics. First, compared to other financial documents like financial news and discussion boards, fund prospectuses are lengthy with hundreds of pages, which increases the time for users to read the documents. Second, fund prospectuses are also difficult to read compared to financial news and discussion boards, since fund prospectuses may use similar sentences to refer to different types of financial concepts. Therefore, it could be helpful for users if the financial concepts could be automatically extracted from fund prospectuses.

III. Contribution

This study explores the significant challenges involved in the automated identification of financial concepts within fund prospectuses, a task that has traditionally required manual intervention by financial regulators. Fund prospectuses, which are essential documents regulated by the SEC, include crucial information about investment funds. However, their complexity and length pose considerable obstacles to automation.

The first challenge arises from the extensive nature of prospectuses, which often results in much longer texts than those typically addressed in existing research. This length complicates the extraction of relevant financial concepts. Additionally, the nuanced language specific to finance means

that even minor variations in wording can lead to significant changes in meaning, making accurate identification even more difficult.

Furthermore, the scarcity of labelled datasets for training machine learning models in this domain presents another critical barrier, limiting the effectiveness of current automated approaches. While existing information extraction methods have proven successful with other document types, such as scientific articles and web pages, they fall short when applied to fund prospectuses.

By tackling these specific challenges, this research seeks to offer valuable insights that will lead to the development of more effective strategies for extracting financial concepts from fund prospectuses. This endeavour aims to enhance regulatory compliance and improve transparency within the financial sector, ultimately establishing a foundation for better information accessibility for all stakeholders involved.

IV. Organization

This study is structured into several key sections to provide a comprehensive exploration of information extraction in financial contexts, as shown in Fig. 2. The Introduction sets the stage by discussing the significance of fund prospectuses within the financial sector and underscores the critical role of information extraction in enhancing regulatory processes. In the Motivation section, the study elaborates on the rationale behind the study, emphasizing the challenges currently faced in automating the extraction of financial concepts from these complex documents. The Organization section outlines the study's structure, guiding readers through the subsequent content. The Contribution section identifies the unique contributions of the research to the field of information extraction, highlighting innovative approaches or insights that advance understanding. Following this, the Information Extraction section provides a general overview of various extraction techniques and their practical applications, laying the groundwork for more specific discussions. The study then shifts focus to Named Entity Recognition (NER), detailing various methods such as rule-based, feature-based, and deep learning-based approaches, and includes a critical discussion on the limitations of existing NER methods.

The Event Extraction section defines the essential concepts and tasks involved in extracting events from financial texts, followed by an examination of both pattern-matching and machine learning-based methods for event extraction. Finally, the study concludes with a section on Research Challenges and Future Directions, which identifies significant challenges in the field and offers recommendations for future research, aiming to inspire further advancements in the extraction of financial information.

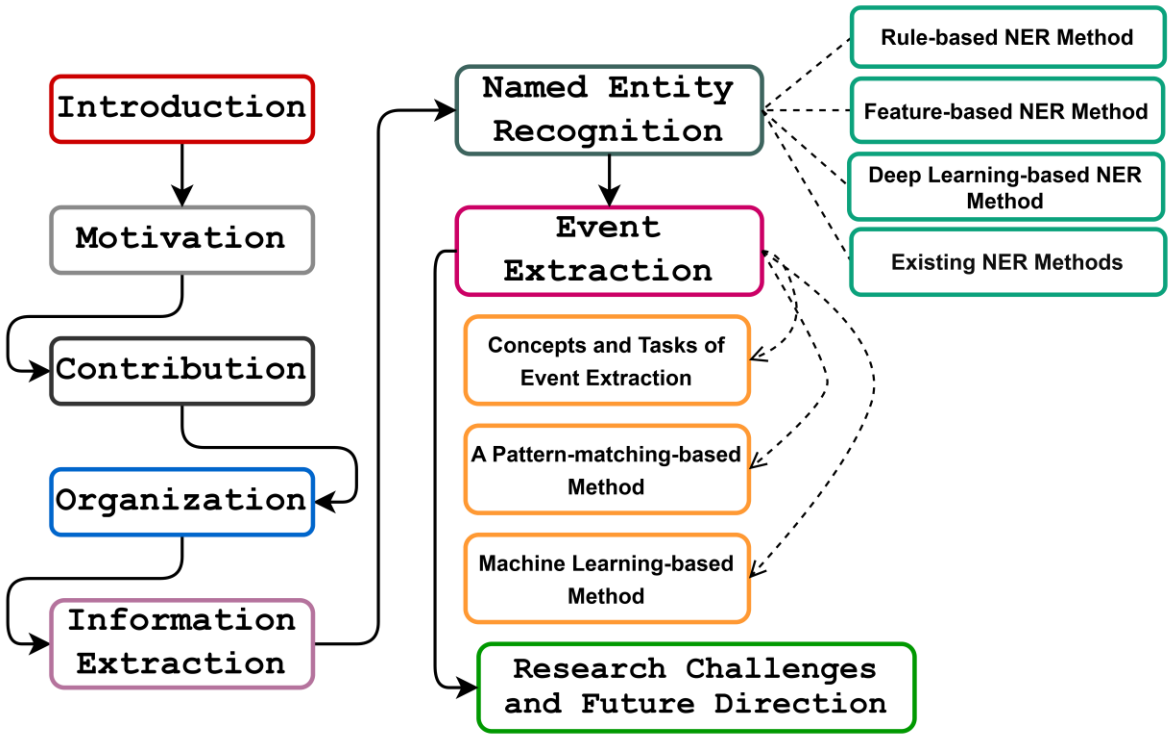


Fig. 2. Composition of the document.

V. Information Extraction

With the swift advancement of information technology, the volume of online data continues to grow daily. Therefore, it becomes the focus of people's attention on how to obtain valuable information from massive data. To achieve this goal, information extraction is the technology for it. Information extraction refers to the structured processing of information contained in text into a specific structure of organization [12]. The typical information extraction tasks include NER and event extraction [13].

- NER refers to the extraction of entities in text; the entities are usually small pieces of phrases describing certain concepts. The example of NER in fund prospectus is the valuation day. The phrase 'Weekly, each Tuesday' explains the day for valuation. However, there are many dates in fund prospectuses; only this date indicates the valuation day, making this entity hard to identify. Another example is 'disclaimer on periodical reports', which should be a link to download the periodical reports. However, there are also many links in the fund prospectuses. Existing NER techniques cannot distinguish them.
- Event extraction is equivalent to the extraction of multiple sentences to explain 5W (i.e., what, who, when, where, why). The example of an event in fund prospectus is a disclaimer on periodical reports. The sentence 'along with the most recent financial reports, all of these documents are available online at nordea.lu' should be extracted as an event, which shows where the periodical reports can be obtained.

For instance, information extraction tasks can retrieve details such as time, location, and key individuals from news articles, as well as product names, development timelines, and performance metrics from technical documents. Additionally, it can extract factual information relevant to users from natural language texts. The applications of information extraction span various fields, including finance, journalism, and healthcare, highlighting its versatility and importance across multiple domains. This subsection defines the tasks of information extraction as follows.

First, for a document with N sentences, D = (S1, S2, ..., SN). Information extraction aims to extract entities, events, and relationships in D. The entity in D is represented as (E1, E2, …, EN) = H (S1, S2, ..., SN), where E1, E2, …, EN are the entities, which are the words in sentences S. NER is to find a function H, what can identify the target E in S. Regarding event extraction, it defines a function P, which can select a set of sentences in S. It has Eventm = Pm (S1, S2, ..., SN), which uses a subset of S to form different events. According to the above formal definition, the task of information extraction is:

"A process of automatically extracting structured information from unstructured and semi-structured data sources. It involves the identification, classification, and extraction of relevant information from textual, numerical, or multimedia data sources" [14].

In this study, the task is to extract the financial concept in fund prospectuses. Financial concepts are the phrases, sentences, and paragraphs to explain the related concepts in financial documents, such as 'fund names, calculation method for issue prices, and disclaimer on periodical reports'. The identification of these concepts is important for regulators to check fund prospectuses. It is also possible to organize fund prospectuses into structured documents based on the extraction results, whose purpose is the same as the purpose of information extraction. Information extraction can be applied to a wide range of domains, including news articles, social media posts, clinical records, financial reports, and scientific papers. The extracted information may include entities (e.g., people, organizations, locations), events (e.g., births, deaths, mergers), and attributes (e.g., product features, customer data).

Information extraction methods are generally categorized according to the type of information being extracted, primarily into NER and event extraction techniques. In this study, this dimension is used for the literature review. The two classes of models are surveyed since this proposal also plans to extract entities (e.g., fund name, website) and events (e.g., the procedure to calculate issue prices) from fund prospectuses.

VI. Deep Learning Network Architecture

The NER (Named Entity Recognition) mainly completes the identification of predefined semantic type named entities from unstructured data. It is currently widely used in search engines, intelligent recommendations, and machine translation. For example, in machine translation, the named entities need to be identified, since each named entity can also have a specified translation.

NER methods can be divided into two methods: rule-based methods and machine learning-based methods [15], as shown in Fig. 3. Rule-based NER methods rely on predefined patterns, dictionaries, and linguistic rules to identify entities in text, making them particularly useful for domain-specific tasks like financial document processing. These methods often use a combination of regular expressions, lexical resources (e.g., lists of financial terms like "management fees" or "dividend yield"), and linguistic rules (e.g., POS tags or dependency parsing) to detect entities. For example, a rule might specify that a sequence of words following the phrase "expense ratio" and preceded by a dollar sign or percentage symbol should be tagged as a financial metric. While rule-based NER is highly interpretable and effective for structured or predictable text, it requires domain expertise to design rules and may struggle with variability in language.

Machine learning-based methods are divided into two categories according to the degree of dependence on the corpus: supervised NER and unsupervised NER. Machine learning-based NER methods use statistical models to automatically learn patterns and features from annotated data for identifying entities in text. These methods typically rely on sequence labelling models, such as Conditional Random Fields (CRFs) or deep learning architectures like BiLSTM-CRF and transformer-based models (e.g., BERT), to predict entity tags (e.g., "ORG," "PER," or "FINANCIAL_TERM") for each token in a sentence. Unlike rule-based methods, ML-based NER does not require manually crafted rules; instead, it learns from labelled datasets, capturing complex linguistic patterns and contextual dependencies. For example, a model trained on financial documents can learn to recognize entities like "management fees" or "dividend yield" based on their context.

This subsection introduces the relevant methods of NER. It also surveys the hot deep learning methods currently studied.

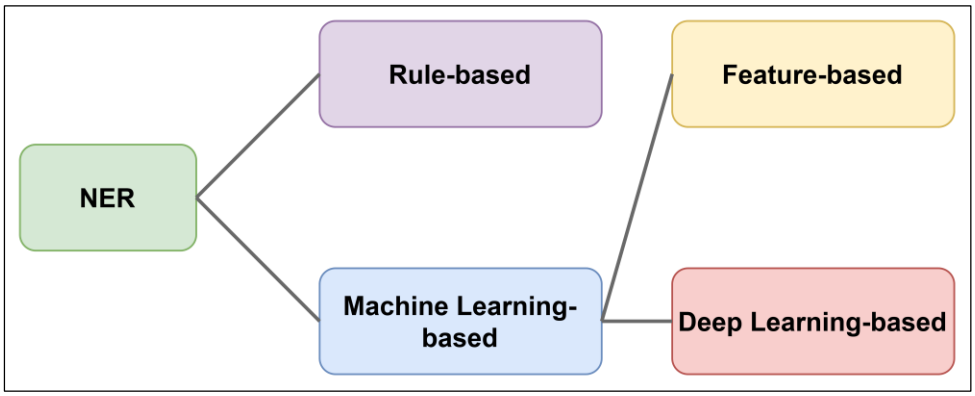


Fig. 3. Classification of NER methods.

A. Rule-Based NER Methods

In the early research stage of NER, the rule-based method was mainly used. The rules are mainly written and formulated by domain experts. The method first writes some simple rules by domain experts, and then experiments are carried out in the corpus. According to [16, 17] the experiment results, the rules are continuously improved after analysing the erroneous results until the effect of NER is satisfactory. Among the rule-based methods, the most commonly used is the dictionary-based matching method, which completes the NER through complete or partial matching of strings. This method is relatively simple and efficient. The implementation of dictionary-based matching methods can usually be implemented by various algorithms, such as forward matching, inverse matching, bidirectional longest matching, and dictionary trees.

The advantage of the rule-based NER method is that there is no need to label the corpus in advance, which has a good effect on a small-scale corpus. The NER system also runs quickly. The disadvantage is that the writing rules have high requirements for the individual level of personnel and poor system portability. Some well-known rule-based NER systems include LaSIE-II [16] and NetOwl [17].

B. Feature-Based NER Method

Feature-based NER methods are supervised methods using traditional machine learning. This method treats NER as a sequence labelling task. The specific algorithm model needs to be trained by using the labelled corpus. In the process of identifying named entities, the feature-based method usually includes the following steps:

a) Corpus labelling: The corpus is typically annotated using either the IOB (Inside-Outside-Beginning) or IO (Inside-Outside) annotation systems, which involve manually labelling the text to identify relevant entities.

b) Feature definition: Key features are defined by selecting the current word, as well as the preceding and following words, along with their parts of speech. These elements significantly influence the performance of NER tasks.

Model Training: Commonly employed models for training. Further, Table I shows several studies on feature-based NER recognition methods, which highlight the implemented methods, target entities, and identify the inadequacies in methods.

TABLE I SUMMARY OF FEATURE-BASED NER METHODS

Method	Description	Target	Type of Documents	Shortcomings
HMM [18]	Analyse word sequence with hidden Markov model	CoNLL2002, CoNLL2003	News and story	These methods demonstrate a notably low level of accuracy.
Traditional CRF [19]	Apply CRF for NER in biomedical domain	CoNLL2002, CoNLL2003	News and story	
CRF-based hybrid model [20]	Combine CRF with rules	CoNLL2002, CoNLL2003	News and story	
Traditional SVM [18]	Apply SVM for NER in biomedical domain	CoNLL2002, CoNLL2003	News and story	
Language independent SVM [21]	A language independent SVM model	CoNLL2002, CoNLL2003	News and story	
Traditional SVM [22]	Apply SVM for NER in real-time text	GENIA	Text in video	

C. Deep Learning-Based NER Method

Compared with the feature-based NER method, the deep learning-based NER method does not require manual rules or complex features. It is easy to extract hidden features from the input corpus. In NER tasks, commonly used neural networks mainly include Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and neural networks based on attention mechanisms. Among them, Long Short-Term Memory Neural Network (LSTM) in RNNs has been widely

used in NER tasks. Li et al., proposed a typical architecture for deep learning NER methods in 2020 [23], as shown in Fig. 4. The architecture is divided into three main parts:

1) Input distributed representations, which consider embeddings at the word and character levels, and combine additional features that are already effective in feature-based methods, such as part-of-speech tags and gazetteers.

2) Context encoder, which uses CNNs, RNNs, or other networks to capture context dependencies.

Tag decoder, which is mainly used to predict the label of the input sequence. Numerous studies on deep learning-based NER methods are presented in Table II, which outlines the employed techniques, target entities, and identifies the shortcomings of each method.

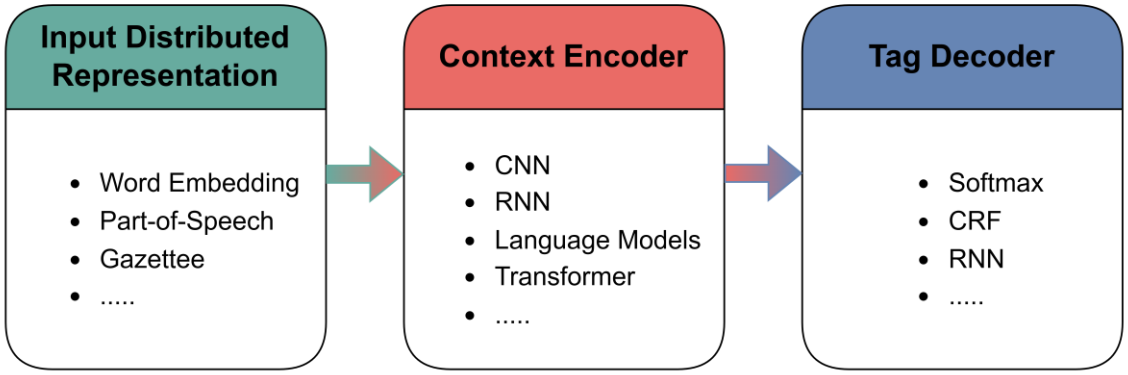


Fig. 4. A typical architecture for deep learning NER methods.

TABLE II SUMMARY OF DEEP LEARNING-BASED NER METHODS

Method	Description	Target	Type of Documents	Shortcomings
BI-LSTM-CRF [24]	A series of LSTM-based sequence labelling models	CoNLL2002, CoNLL2003 (Sentence)	News and story	These methods are intended to extract concise phrases from lengthy sentences (e.g., names, dates, and locations). However, they are built on a sentence-level model, and the target corpora do not pertain to the financial domain.
BiLSTM-CNNs-CRF [25]	An end-to-end sequence labeling model that combines bidirectional LSTM, CNN, and CRF	WSJ, CoNLL2003 (Sentence)	Newspaper, news, and story	
Attention-RNN-CRF [26]	A combination method	CoNLL2000, CoNLL2003, PTB-POS (Sentence)	News and story	
LSTM-CRF [27]	A method that uses LSTM and CRF	CoNLL2000, CoNLL2003 (Sentence)	News and story	
MRC-NER [28]	A method based on machine reading comprehension	WSJ, CoNLL2003 (Sentence)	Newspaper, news, and story	
TENER [29]	An improved Transformer encoder to model character-level and word-level features	CoNLL2003, OntoNotes 5.0 (Sentence)	News, story, blogs, conversations	
SDI-NER [30]	Combining syntactic dependency graphs for Chinese NER	Weibo	Blogs, short, messages	
WNER [31]	Using word-word relations for NER	CoNLL2003, Onto Notes 5.0, Weibo (Sentence)	News, story, blogs, short messages, conversations	
Survey [32]	A survey of NER using deep learning	NaN	NaN	
MINER [33]	Improving Out-of-Vocabulary Named Entity Recognition from an information-theoretic perspective	WNUT2017, TwitterNER, BioNER, Conll03-Typos, Conll03-OOV (Sentence)	Short messages, News, technical literature, Blogs	
kNN-NER [34]	NER with nearest neighbour search	CoNLL2003, OntoNotes,5.0OntoNotes, 4.0, MSRA, Weibo (Sentence)	Short messages, News, story	

E-NER [35]	Evidential deep learning for trustworthy NER	CoNLL2003, OntoNotes 5.0, WikiGold, TwitterNER, CoNLL2003-Typos, CoNLL2003-OOV (Sentence)	Short messages, News, story, Encyclopaedia	
GPT-NER [36]	NER via large language models	CoNLL2003, OntoNotes5.0 (Sentence)	News, short messages, story	
PromptNER [9]	Prompting for NER	CoNLL2003, CrossNER, GENIA (Sentence)	News, story, Biomedical terminology	
UniversalNER [37]	Targeted distillation from large language models for open NER	ACEO5, AnatEM, bc2gm, bc4chemd, bc5cdr, Broad Twitter, CoNLLO3, FabNER, Findvehice, GENIA, HarveyNER	News, short messages, story, Blogs etc.	

Huang et al. proposed a series of LSTM-based sequence labelling models, including LSTM, Bidirectional LSTM (BI-LSTM), LSTM (LSTM-CRF) with Conditional Random Field (CRF), and Bidirectional LSTM (BI-LSTM-CRF) with CRF layer, and compared the performance of the above models on NLP-labelled datasets [24]. In their work, the BI-LSTMCRF model is applied to the NLP benchmark sequence labelling dataset for the first time, and it is proven that the model can effectively utilize past and future input features and sentence-level labelling information. Ma et al. introduced an end-to-end sequence labelling model that integrates bidirectional LSTM, CNN, and CRF [25]. This model operates independently of task-specific resources, feature engineering, and data pre-processing, making it highly versatile.

Rei et al. (2016) proposed a combination method of word vector and character-level vector based on an attention mechanism for sequence labelling tasks [26]. This method argues that NER requires not only word vectors, but also character-level feature vectors in words. Based on the RNN-CRF model, the attention mechanism is used to splice word vectors and character-level feature vectors. Lample et al. (2016) proposed the LSTM-CRF NER model, which relies on two sources of information about words: character-based word representations are mainly learned from supervised corpora; Unsupervised word representation is learned primarily from an unannotated corpus [27].

Li et al. (2019) proposed a framework based on Machine Reading Comprehension (MRC) [28]. This method works with both non-nested and nested types of NER. Compared with the sequence annotation method, this method is simple and intuitive, and has strong portability. Experiments show that the MRC-based method can encode some prior semantic knowledge of the problem, so that it can perform better under small data sets and transfer learning. Yan, Deng, Li, & Qiu (2019) introduced the TENER model, which enhances the original Transformer architecture for NER tasks [29]. This model utilizes an improved Transformer encoder to effectively capture both character-level and word-level features.

Recent advancements in NER have seen diverse methodological innovations. Li et al. (2022) conducted a comprehensive survey highlighting the evolution of deep learning techniques in NER, providing a foundation for contemporary research [32]. Wang et al. (2022) introduced MINER, addressing out-of-vocabulary entity recognition through information-theoretic principles, while their kNN-NER leveraged nearest neighbour search to enhance contextual similarity modelling [33]. Zhang et al. (2023) proposed E-NER, integrating evidential deep learning to quantify uncertainty and improve trustworthiness in predictions [35]. The rise of large language models (LLMs) spurred novel approaches: Wang et al. (2023) developed GPT-NER [36], demonstrating LLMs' potential for direct entity recognition, while Ashok and Lipton (2023) introduced PromptNER [9], optimizing task-specific prompting strategies. Zhou et al. (2023) advanced UniversalNER [37], employing targeted distillation from LLMs to achieve robust performance in open-domain settings. Together, these works reflect a dynamic shift toward leveraging LLMs, probabilistic frameworks, and innovative training paradigms to tackle NER challenges across diverse contexts.

VII. Discussion on Existing NER Methods

As depicted in the last column of Table I, the existing information extraction techniques for NER are not suitable for financial concept extraction in fund prospectuses. The first reason is that all these methods are designed to extract short phrases from sentences, such as person names, URLs, times, and places. For example, a commonly used dataset for training and testing these methods is a dataset such as the CoNLL2003 dataset for NER tasks. CoNLL2003 covers English and German, including around 14,000 documents for English and 12,000 for German, with an average of 12 to 18 sentences per document. However, the financial concepts in fund prospectuses are usually associated with long sentences or paragraphs. These methods cannot model such sentence-level information to extract. The second reason is that, for machine learning based methods, they usually require a large dataset to train or fine-tune the model (e.g., the deep learning methods). As discussed previously, it is not easy to manually label such datasets for financial concept extraction; hence, the effectiveness of these methods could be limited [2].

This study takes the work by Wang et al. (2022) as a baseline [33, 34]. The work proposes MINER to address out-of-vocabulary entity recognition through information-theoretic principles. Since financial documents are different from many existing NER datasets (e.g., news), many out-of-vocabulary entities could exist. MINER can be a good baseline to address this issue. On top of MIER, this proposal represents domain

expert rules as additional information to improve the ability of MINER in identifying long entities (i.e., financial concepts).

VIII. Event Extraction

An event is an objective fact consisting of a specific person, object, and action at a specific time and place. As an important part of information extraction, the key to event extraction is to extract information about events of interest from unstructured text and store it in a structured way [38]. This subsection first introduces the concept and tasks of event extraction. Then, the related works of event extraction based on pattern matching, machine learning, and neural networks are summarized.

A. Concepts and Tasks of Event Extraction

Gathering Event extraction techniques involve concepts such as triggers, event arguments, and argument roles, as listed in Table III.

TABLE III Basic Concepts and Terms for Event Extra

Concepts	Definition
Trigger	The rigger is a word. It is usually a verb or nouns and phrases that represent an action
Event arguments	The entity and entity property that are related to the event, including time, place, and persons
Argument roles	The role of the entity in the event, namely, the relationship between the entity and the event

The aim of event extraction is to identify the type of event and automatically extract arguments with varying roles from unstructured text using machine learning techniques [39]. The event extraction task is divided into two parts: Event Detection (ED) and Argument Extraction (EAE). Among them, event detection aims to identify specific types of event trigger words from a given text; it is a key step in event extraction [40]. Event Element Extraction identifies the elements of a particular event and marks its role [41]. For example, in the sentence “Biden swears inauguration at Capitol on January 20”, the event extraction task is specifically to detect the trigger word “inauguration”, which is the incident type; The words "Biden", "January 20" and "Capitol" are as the event elements, which can be determined as "person", "time" and "place", respectively.

The implementation of the above event extraction task is based on the technical basis of word segmentation, part-of-speech, and propositional entity recognition [42, 43]. The extracted structured information can further support deeper mining and understanding of unstructured text.

Table IV summarizes the main methods for event extraction. The methods can be classified into pattern-matching-based methods and machine learning-based methods. Early event extraction techniques typically use a pattern-matching-based method. Pattern-matching-based event extraction involves identifying specific events in text by matching predefined patterns or templates that describe how events are typically expressed.

TABLE IV Summary of Event Extraction Methods

Methods		Characteristics
Pattern matching-based		Works better in specific areas, but the process is cumbersome, prone to cascading errors, and depends on the specific text form of the specific domain
Machine learning-based	Feature-based	Relies on complex feature engineering and natural language processing tools
	Neural network-based	Works better in many EE tasks. It requires large corpus to train or fine-tune the model. It takes a long time to train the model and conduct prediction

These patterns often combine lexical cues (e.g., keywords like "acquire" or "merge"), syntactic structures (e.g., verb-argument relationships), and semantic constraints (e.g., entity types involved in the event). Pattern-matching methods are highly interpretable and effective for structured or predictable text, such as news articles or financial reports, where events follow consistent linguistic patterns. However, they require domain expertise to design patterns and may struggle with linguistic variability or complex sentence structures.

In recent years, due to the promotion of the construction of large-scale corpus and the rise of machine learning, researchers have begun to explore methods based on machine learning. Machine learning-based event extraction methods use statistical models to automatically identify and classify events (e.g., mergers, earnings announcements) in text by learning patterns from annotated data. These approaches often frame event extraction as a structured prediction task, where models like sequence labelling (e.g., BiLSTM-CRF) or transformers (e.g., BERT, GPT) are trained to detect event triggers (e.g., verbs like "acquire" or "announce") and their associated arguments (e.g., entities, time, location). For instance, a model might learn that the phrase "Company X announced a merger with Company Y" contains a "merger" event involving two organizations. Unlike rule-based methods, ML approaches generalize better to linguistic variability and implicit event mentions (e.g., "the acquisition was finalized"). They leverage contextual embeddings to capture semantic relationships and dependencies, but require large, labelled datasets for training.

B. A Pattern-Matching-Based Method

The pattern-matching-based method consists of two steps: pattern acquisition and event extraction. Firstly, the event pattern extraction is obtained through local text analysis, such as lexical analysis and syntactic analysis. Then, under the guidance of the event pattern, the event sentence to be extracted is matched to the corresponding pattern to detect and extract an event type, as shown in Fig. 5.

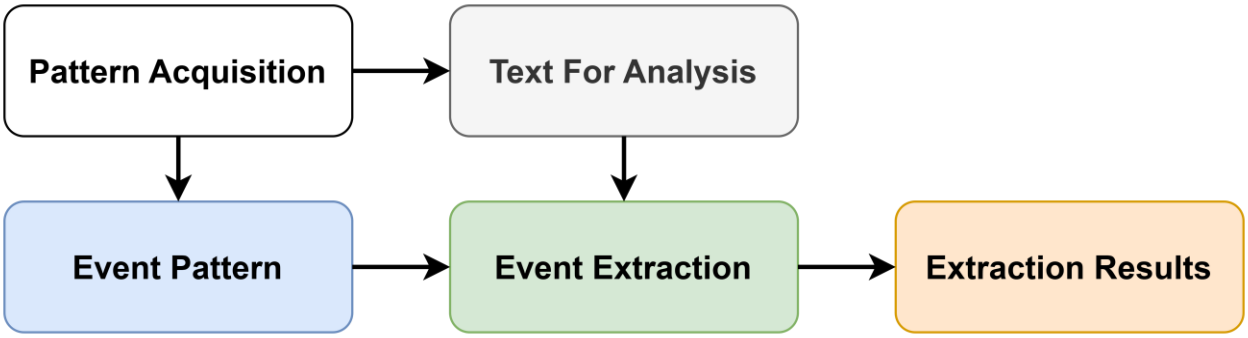


Fig. 5. Event extraction method based on pattern matching.

The supervised pattern matching method relies on a manually labelled corpus. The quality of the human-labelled corpus affects the effect of pattern learning. Therefore, Riloff (1993) mentions that statements near event elements often contain descriptions of the role of event elements in events [44]. This work constructs a domain-specific pattern for event extraction through supervised learning combined with human review.

Supervised pattern matching methods rely heavily on labelled corpus. Therefore, some researchers propose methods based on weak supervision. This type of method only requires manual pre-classification of the corpus or the definition of a small number of artificial seed patterns. Then, the method automatically learns the event mode according to the pre-classified corpus or seed mode. Based on the literature, the process only needs to label the event type but does not need to label all event elements in the corpus, which reduces the workload required to create a corpus. Table V summarizes the main works of pattern-based event extraction. A previous study (Roman, 2000) constructs a seed pattern set, divides the unlabelled corpus into related text sets and unrelated text sets [45]. It then explores the relevant patterns of related text sets to learn new models incrementally. Although this method (Roman, 2000) does not require artificial corpus labelling, the pre-classification labelling and seed pattern set they introduce are still relatively heavy. In order to solve this problem, the paper (Jiang, 2005) proposes a domain-independent pattern representation method, which is integrated into the WordNet conceptual knowledge base [46]. When performing pattern acquisition, this method only needs to define the event extraction task. The corresponding event extraction pattern can be automatically learned from the original corpus, which minimizes the intervention of manual annotation on pattern-matching event extraction. Other early studies, such as [47], demonstrated the utility of domain-specific patterns in biomedical event extraction, leveraging tailored rules to extract events from scientific literature. Similarly, Liao and Grishman (2011) combined semantic role labelling (SRL) with pattern-based rules to refine event extraction, showcasing the synergy between linguistic structures and rule-based methods. In recent years, pattern-based approaches have been enhanced by integrating them with deep learning techniques. For instance, Zhang et al. (2020) proposed a rule-augmented deep learning framework for event extraction in low-resource settings, emphasizing the value of pattern-based methods when annotated data is scarce [41]. Chen et al. (2021) further advanced this hybrid approach by incorporating dynamic multi-pooling convolutional neural networks (CNNs) to improve contextual understanding and event detection accuracy [48]. Additionally, Li et al. (2022) explored the use of transformer models like BERT alongside pattern-based rules for clinical event extraction, demonstrating their effectiveness in specialized domains [49]. These studies collectively illustrate the evolution of pattern-based methods, from standalone rule-based systems to hybrid frameworks that leverage the strengths of both traditional and modern NLP techniques. Learning models utilized for UATR often exhibit high complexity, which can lead to significant challenges in training effectiveness. When the complexity of a model exceeds the available quantity and quality of training data, achieving desired outcomes becomes increasingly difficult. While improving model accuracy is a primary objective, relying solely on accuracy as the sole metric for evaluating model performance is insufficient.

TABLE V SUMMARY OF PATTERN-BASED METHODS

Reference	Contribution	Usage scenarios
[45]	Build a seed pattern set for event extraction	Pattern-based methods require constructing patterns to extract events; however, the pattern is not easy to build, which limits their usage scenarios.
[46]	Propose domain-independent patterns	
[47]	Propose domain-specific patterns	
[50]	Combined semantic role labelling with pattern-based rules	
[41]	Propose a rule-augmented deep learning framework for event extraction	
[48]	A hybrid approach by incorporating dynamic multi-pooling CNNs	
[49]	Explored the use of transformer	

Although the pattern-based matching method is more effective in specific domains, the method is cumbersome in the process, prone to cascading errors, and depends on the specific form of the domain-specific text and requires participants to have strong professionalism in the process of obtaining patterns. Hence, its cost is high, and the formulated pattern is difficult to cover all event types. When the corpus changes, the pattern also needs to be updated, resulting in poor system generalization and low recall.

C. Machine Learning-Based Method

Compared with pattern matching methods, machine learning methods are more adaptable and portable to different domains. The machine learning-based method transforms the event extraction task into a classification problem on the basis of statistical learning and selects the appropriate feature input to the classifier to complete the extraction task. Typically, these classifiers include event trigger word classifiers, argument classifiers, and argument role classifiers. These classifiers can employ classification models such as maximum entropy and support vector machines in machine learning.

Chieu et al. first time applies the maximum entropy model for argument extraction and constructs a classifier to realize the event extraction of lecture announcements by defining simple features such as named entities, first words in sentences, lexical lowercase forms, and time expressions [51]. In order to extract more features in the feature engineering stage and improve performance, Llorens et al. (2010) also introduce semantic corner colour features; it use conditional random field model to realize event extraction and achieve good results [52]. Li et al. (2013) [53] further introduce global features to improve performance on the basis of Llorens et al. (2010) [52], and use trigger words and dependencies between multiple arguments for event extraction.

Different machine learning algorithms often have different advantages and disadvantages. It is possible to combine multiple machine learning methods to improve the event extraction effect. Ahn (2006) combines two machine learning models, which are the memory-based nearest neighbour learner and maximum entropy learner, to propose a simple modular event extraction method [54]. This method constructs multiple classifiers for each module separately; it uses lexical features, dictionary features, context features, dependency features, and entity features to complete the event extraction.

Machine learning-based event extraction often relies on complex feature engineering and natural language processing tools, which often require deep exploratory data analysis on the corpus. The cumbersome and tedious nature of this process, as well as the accumulation and propagation of errors caused by various types of feature extraction processes, will more or less affect the extraction results.

In recent years, deep learning has been a popular method to represent data in machine learning. Compared with traditional machine learning methods (feature-based), deep learning avoids the tedious work of manual feature engineering; it does not require rich domain expert knowledge and only needs to pass data directly to the built network. Its portability and flexibility have prompted more and more researchers to focus on deep learning-based methods.

The event extraction method based on deep learning can be divided into the method of pipeline model and the method of a document-level model, according to the different learning methods of the model. Pipeline model is a model that recognizes named entities and extracts events as a pipeline, while the method of document-level models is to learn the characteristics of events for extracting events directly without considering the named entities. Depending on the range of features used, it can be divided into sentence-level methods and chapter-level methods. Table VI displays recent works that utilize a combination of pipeline and document-level models, with the majority employing deep learning-based approaches.

TABLE VI SUMMARY OF MACHINE LEARNING-BASED METHODS

Ref	Contribution	Usage scenarios
[11]	Using regression forest for event extraction	Extract events in two steps. First it extracts named entities, and then events are extracted based on the named entities. As explain in existing methods for NER may not work on fund prospectus, which can affect the effectiveness of applying pipe models on financial concept extraction.
[55]	Ensemble learning for event extraction	
[56]	SVM based event extraction methods	
[57]	Event time extraction with decision tree	
[58]	Relation extraction for imbalanced data	
[48]	proposes a dynamic multi-pool convolutional network	
[59]	propose a framework based on the pre-trained model BERT	
[60]	uses LSTM (LongShort-Term Memory) for event detection	
[61]	use Bi-LSTM model	
[62]	uses the pre-trained model BERT to fine-tune Bi-LSTM	
[63]	adds an attention mechanism after the Bi-LSTM layer	
[64]	Add a gate mechanism in each graph convolutional layer	
[65]	Propose the edge-enhanced graph convolutional network	
[66]	Combine the syntactic and semantic structures through the application of the graph converter network	
[67]	Combine knowledge from hierarchical knowledge graphs and graphs neural networks on the basis of pre-training models	
[68]	Combine coreference resolution to better identify the event	
[49]	A survey for event extraction using deep learning	
[69]	Event Detection as Multi-task Text Generation	
[10]	A Data-Efficient Generation-Based Event Extraction Model	
[70]	Boosting Low-Resource Information Extraction by Structure-to-Text Data Generation with Large Language Models	

The event extraction method based on the pipeline model executes subtasks sequentially. Subsequent tasks depend on the results of the former. Chen at el. (2015) present event extraction as a two-stage multi-classification process, introducing a dynamic multi-pool convolutional network [48]. In the first stage, each word is classified to detect a trigger word. If a trigger is identified, the second stage classifies the trigger and assigns its meta-role. Yang at el. (2019) propose a framework based on the pre-trained model BERT (Bidirectional Encoder Representations Transformers), which includes a trigger word classifier and a metaclassifier that obtains results by inference and referencing the former [59].

Different classifiers can also be used for each stage. For example, Ding at el. (2019) uses LSTM (LongShort-Term Memory) in the event detection stage to construct a trigger perceptual feature extractor to dynamically learn character-level, part-of-speech, and semantic-level features to achieve event detection on the Chinese corpus [60]. Compared with the LSTM model, Bi-LSTM (Bi-directional Long Short-Term Memory) can learn long-term dependent information more effectively in sequence labelling tasks. Zeng et al. (2016) use a Bi-LSTM model to capture the semantic and structural features of sentences and achieve good results in the event detection stage [61]. Satyapanich et al. (2020) use the pre-trained model BERT to fine-tune Bi-LSTM in the event detection stage for trigger word classification [62], while the method of Zheng et al. (2018) in the argument extraction stage adds an attention mechanism after the Bi-LSTM layer for meta-recognition and finally assigns the meta role through the neural network [63].

In the deep neural network, hidden vectors in graph convolutional networks may contain noise information that is not related to candidate trigger words. Dac Lai et al. (2020) add a gate mechanism in each graph convolutional layer to filter the noise information in the hidden vector, but this method ignores the dependent label information [64]. The edge-enhanced graph convolutional network proposed by Cui et al. (2020) has achieved good results on the ACE2005 dataset by combining syntactic structure and type-dependent labels to improve event detection [65]. In order to capture the intermediate structure of the sentence, Veyseh et al. (2020) combine the syntactic and semantic structures through the application of the graph converter network; it captures the intermediate structure of the sentence from different input perspectives and then multiplies the intermediate structure to generate the final structure [66]. In addition, in order to enhance the inference ability of complex events, Li et al. (2019) combine knowledge from hierarchical knowledge graphs and graph neural networks on the basis of pre-training models [67].

Recent advancements in event extraction have introduced innovative approaches to address its inherent challenges. Lu et al. (2022) enhanced event identification by integrating coreference resolution, improving the ability to link event mentions across text [68]. Li et al. (2022) provided a comprehensive survey of deep learning techniques in event extraction, offering a roadmap for ongoing research [32]. Anantheswaran at el. (2023) reimagined event detection as a multi-task text generation problem, leveraging generative models to capture complex event structures [69]. Hsu et al. (2023) proposed a data-efficient generation-based model, addressing the scarcity of labelled data by optimizing generative frameworks for event extraction [10]. Ma et al. (2023) tackled low-resource scenarios by employing large language models for structure-to-text data generation, significantly boosting performance in information extraction tasks [70]. Together, these works highlight a shift toward leveraging generative models, coreference resolution, and data-efficient strategies to advance event extraction capabilities across diverse domains.

IX. Research Challenges and Future Directions

Event extraction extracts events from documents. The event is usually longer than named entities; hence, this task is more similar than financial concept extraction. Currently, many methods have been proposed. They extract events in two steps. First, it extracts named entities, and then events are extracted based on the named entities. As discussed previously, the existing methods for NER may not work on a fund prospectus, which can affect the effectiveness of applying pipe models on financial concept extraction. Second, there are ambiguity problems in fund prospectuses. Similar sentences may have different meanings when the context is different. The same entity can also have different meanings in different contexts, as discussed previously. Therefore, when extracting information, it is important to consider the context of the event (i.e., sentences) and conduct a multi-grained comparison on sentences (e.g., on word-level, sentence-level, keyword-level). Third, all these methods require a large, labelled data set to train the model, which is not available in financial concept extraction.

A. Small Training Set Issue in Financial Document Extraction Domain

Due to the scarcity of labelled training data, financial documents are often highly specialized, containing domain-specific terminology, complex structures, and sensitive information, making it difficult to obtain large, high-quality annotated datasets. There are a few studies to address the small training set issue in financial document information extraction.

B. Data Augmentation

The first solution is data augmentation. Anantheswaran et al. introduced a multi-task text generation framework for event detection, which can be adapted to financial information extraction [69]. By generating diverse textual representations of financial events, their method effectively expanded the training set without requiring additional manual annotations.

C. Transferring Learning and Pre-Trained Language Models

The sconed solution is transferring learning and pre-trained language models. Hsu et al. developed a data-efficient generation-based model for event extraction, leveraging pre-trained language models (PLMs) to transfer knowledge from general-domain corpora to financial documents [10]. Their approach minimized the need for large, annotated datasets by fine-tuning PLMs on small, domain-specific data.

D. Few-Shot and Zero-Shot Learning

The third solution is few-shot and zero-shot learning. Ashok and Lipton introduced PromptNER, a prompting-based approach for NER that enables few-shot and zero-shot learning [9]. By framing information extraction tasks as natural language prompts, their method reduced reliance on large, annotated datasets, making it particularly suitable for financial documents with limited labelled examples. Zhou et al. proposed UniversalNER, which uses targeted distillation from LLMs to perform open-domain NER with minimal supervision [37]. Their approach demonstrated robustness in low-resource settings, including financial document information extraction.

As shown in Table IV and Table V, there are many publicly available datasets for NER such as CoNLL2003. However, these datasets do not have financial concepts to extract. For

example, the CoNLL2003 dataset is widely used benchmarks for NER tasks. CoNLL2003 covers English and German. CoNLL2003, published in 2003, includes around 14,000 documents for English and 12,000 for German, with an average of 12 to 18 sentences per document. The dataset primarily consists of news articles, making them representative of the news domain. Therefore, these datasets cannot solve the small training set issue in financial concept extraction.

X. Conclusion

In conclusion, the comprehensive analysis of information extraction techniques in this study highlights the critical importance of accurately capturing financial concepts within fund prospectuses. As established, the completeness of content in these documents is vital for investment funds, necessitating rigorous oversight from financial regulators. Despite the availability of various information extraction methods, significant challenges remain when applying these techniques to fund prospectuses, particularly due to their inherent complexity, linguistic ambiguity, and the scarcity of labelled data for effective training. The identified gaps in existing research emphasize the need for tailored frameworks specifically designed for financial concept extraction in this context. This research addresses these specific challenges to provide valuable insights that will facilitate the creation of more effective strategies for extracting financial concepts from fund prospectuses. The goal is to enhance regulatory compliance and increase transparency in the financial sector, thereby laying the groundwork for improved information accessibility for all stakeholders.

References

[1] H. Li, Q. Yang, Y. Cao, J. Yao, and P. Luo, "Cracking tabular presentation diversity for automatic cross-checking over numerical facts," in Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2020, pp. 2599-2607.

[2] A. Gupta, V. Dengre, H. A. Kheruwala, and M. Shah, "Comprehensive review of text-mining applications in finance," Financial Innovation, vol. 6, no. 1, p. 39, 2020.

[3] Y. Arslan et al., "A comparison of pre-trained language models for multi-class text classification in the financial domain," in Companion proceedings of the web conference 2021, 2021, pp. 260-268.

[4] M. Ceci, D. Bianculli, and L. C. Briand, "Defining a model for content requirements from the law: An experience report," in 2024 IEEE 32nd International Requirements Engineering Conference (RE), 2024, pp. 18-30: IEEE.

[5] Q. Li, Y. Chen, J. Wang, Y. Chen, and H. Chen, "Web media and stock markets: A survey and future directions from a big data perspective," IEEE Transactions on Knowledge and Data Engineering, vol. 30, no. 2, pp. 381-399, 2017.

[6] Q. Li, S. Shah, and R. Fang, "Table classification using both structure and content information: A case study of financial documents," in 2016 IEEE International Conference on Big Data (Big Data), 2016, pp. 1778-1783: IEEE.

[7] J. Cleland-Huang, O. C. Gotel, J. Huffman Hayes, P. Mäder, and A. Zisman, "Software traceability: trends and future directions," in Future of software engineering proceedings, 2014, pp. 55-69.

[8] R. Grishman, "Twenty-five years of information extraction," Natural Language Engineering, vol. 25, no. 6, pp. 677-692, 2019.

[9] D. Ashok and Z. C. Lipton, "Promptner: Prompting for named entity recognition," arXiv preprint arXiv:2305.15444, 2023.

[10] I. Hsu et al., "DEGREE: A data-efficient generation-based event extraction model," arXiv preprint arXiv:2108.12724, 2021.

[11] H. Phan, M. Maaß, R. Mazur, and A. Mertins, "Random regression forests for acoustic event detection and classification," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 23, no. 1, pp. 20-31, 2014.

[12] G. R. Doddington, A. Mitchell, M. A. Przybocki, L. A. Ramshaw, S. M. Strassel, and R. M. Weischedel, "The automatic content extraction (ace) program-tasks, data, and evaluation," in Lrec, 2004, vol. 2, no. 1, pp. 837-840: Lisbon.

[13] R. Grishman and B. M. Sundheim, "Message understanding conference-6: A brief history," in COLING 1996 volume 1: The 16th international conference on computational linguistics, 1996.

[14] K. Adnan and R. Akbar, "An analytical study of information extraction from unstructured and multidimensional big data," Journal of Big Data, vol. 6, no. 1, pp. 1-38, 2019.

[15] D. Nadeau and S. Sekine, "A survey of named entity recognition and classification," Lingvisticae Investigationes, vol. 30, no. 1, pp. 3-26, 2007.

[16] K. Humphreys et al., "University of Sheffield: Description of the LaSIE-II system as used for MUC-7," in Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29-May 1, 1998, 1998.

[17] G. Krupka and K. Hausman, "IsoQuest Inc.: description of the NetOwl™ extractor system as used for MUC-7," in Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29-May 1, 1998, 1998.

[18] S. Morwal, N. Jahan, and D. Chopra, "Named entity recognition using hidden Markov model (HMM)," International Journal on Natural Language Computing (IJNLC) Vol, vol. 1, 2012.

[19] U. Kanimozhi and D. Manjula, "A CRF based machine learning approach for biomedical named entity recognition," in 2017 second international conference on recent trends and challenges in computational models (ICRTCCM), 2017, pp. 335-342: IEEE.

[20] Z. Xu, X. Qian, Y. Zhang, and Y. Zhou, "CRF-based hybrid model for word segmentation, NER and even POS tagging," in Proceedings of the sixth SIGHAN workshop on Chinese language processing, 2008.

[21] A. Ekbal and S. Bandyopadhyay, "Named entity recognition using support vector machine: A language independent approach," International Journal of Electrical, Computer, and Systems Engineering, vol. 4, no. 2, pp. 155-170, 2010.

[22] A. T. Imam, A. Alhroob, and W. Alzyadat, "SVM machine learning classifier to automate the extraction of SRS elements," International Journal of Advanced Computer Science and Applications (IJACSA), 2021.

[23] J. Li, A. Sun, J. Han, and C. Li, "A survey on deep learning for named entity recognition," IEEE transactions on knowledge and data engineering, vol. 34, no. 1, pp. 50-70, 2020.

[24] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF models for sequence tagging," arXiv preprint arXiv:1508.01991, 2015.

[25] X. Ma and E. Hovy, "End-to-end sequence labeling via bi-directional lstm-cnns-crf," arXiv preprint arXiv:1603.01354, 2016.

[26] M. Rei, G. K. Crichton, and S. Pyysalo, "Attending to characters in neural sequence labeling models," arXiv preprint arXiv:1611.04361, 2016.

[27] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition," arXiv preprint arXiv:1603.01360, 2016.

[28] X. Li, J. Feng, Y. Meng, Q. Han, F. Wu, and J. Li, "A unified MRC framework for named entity recognition," arXiv preprint arXiv:1910.11476, 2019.

[29] H. Yan, B. Deng, X. Li, and X. Qiu, "TENER: adapting transformer encoder for named entity recognition," arXiv preprint arXiv:1911.04474, 2019.

[30] P. Zhu et al., "Improving Chinese named entity recognition by large-scale syntactic dependency graph," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 30, pp. 979-991, 2022.

[31] S. Ali, K. Masood, A. Riaz, and A. Saud, "Named entity recognition using deep learning: A review," in 2022 International Conference on Business Analytics for Technology and Security (ICBATS), 2022, pp. 1-7: IEEE.

[32] J. Li et al., "Unified named entity recognition as word-word relation classification," in proceedings of the AAAI conference on artificial intelligence, 2022, vol. 36, no. 10, pp. 10965-10973.

[33] X. Wang et al., "MINER: Improving out-of-vocabulary named entity recognition from an information theoretic perspective," arXiv preprint arXiv:2204.04391, 2022.

[34] S. Wang et al., "$ k $ NN-NER: Named Entity Recognition with Nearest Neighbor Search," arXiv preprint arXiv:2203.17103, 2022.

[35] Z. Zhang et al., "E-ner: evidential deep learning for trustworthy named entity recognition," arXiv preprint arXiv:2305.17854, 2023.

[36] S. Wang et al., "Gpt-ner: Named entity recognition via large language models," arXiv preprint arXiv:2304.10428, 2023.

[37] W. Zhou, S. Zhang, Y. Gu, M. Chen, and H. Poon, "Universalner: Targeted distillation from large language models for open named entity recognition," arXiv preprint arXiv:2308.03279, 2023.

[38] G. Xiyue and H. Tingting, "Survey about research on information extraction," Computer science, vol. 42, no. 2, pp. 14-17, 2015.

[39] Z. Kan, L. Qiao, S. Yang, F. Liu, and F. Huang, "Event arguments extraction via dilate gated convolutional neural network with enhanced local features," IEEE Access, vol. 8, pp. 123483-123491, 2020.

[40] S. Liu, Y. Chen, S. He, K. Liu, and J. Zhao, "Leveraging framenet to improve automatic event detection," in Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2016, pp. 2134-2143.

[41] N. Zhang et al., "Document-level relation extraction as semantic segmentation," arXiv preprint arXiv:2106.03618, 2021.

[42] Y. Tian et al., "Joint Chinese word segmentation and part-of-speech tagging via two-way attentions of auto-analyzed knowledge," in Proceedings of the 58th annual meeting of the association for computational linguistics, 2020, pp. 8286-8296.

[43] R. Aly, A. Vlachos, and R. McDonald, "Leveraging type descriptions for zero-shot named entity recognition and classification," in Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2021, pp. 1516-1528.

[44] E. Riloff, "Automatically constructing a dictionary for information extraction tasks," in AAAI, 1993, vol. 1, no. 1, p. 2.1.

[45] R. Yangarber, Scenario customization for information extraction. New York University, 2000.

[46] J. Jiang, "An event IE pattern acquisition method," Computer Engineering, vol. 31, no. 15, pp. 96-98, 2005.

[47] J. Björne and T. Salakoski, "Biomedical event extraction using convolutional neural networks and dependency parsing," in Proceedings of the BioNLP 2018 workshop, 2018, pp. 98-108.

[48] Y. Chen, L. Xu, K. Liu, D. Zeng, and J. Zhao, "Event extraction via dynamic multi-pooling convolutional neural networks," in Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2015, pp. 167-176.

[49] J. Li et al., "Automatic text classification of actionable radiology reports of tinnitus patients using bidirectional encoder representations from transformer (BERT) and in-domain pre-training (IDPT)," BMC Medical Informatics and Decision Making, vol. 22, no. 1, p. 200, 2022.

[50] S. Liao and R. Grishman, "Acquiring topic features to improve event extraction: in pre-selected and balanced collections," in Proceedings of the international conference recent advances in natural language processing 2011, 2011, pp. 9-16.

[51] H. L. Chieu and H. T. Ng, "A maximum entropy approach to information extraction from semi-structured and free text," Aaai/iaai, vol. 2002, pp. 786-791, 2002.

[52] H. Llorens, E. Saquete, and B. Navarro, "TimeML events recognition and classification: learning CRF models with semantic roles," in Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), 2010, pp. 725-733.

[53] Q. Li, H. Ji, and L. Huang, "Joint event extraction via structured prediction with global features," in Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2013, pp. 73-82.

[54] D. Ahn, "The stages of event extraction," in Proceedings of the Workshop on Annotating and Reasoning about Time and Events, 2006, pp. 1-8.

[55] R. Xu, J. Hu, Q. Lu, D. Wu, and L. Gui, "An ensemble approach for emotion cause detection with event extraction and multi-kernel svms," Tsinghua Science and Technology, vol. 22, no. 6, pp. 646-659, 2017.

[56] Y. Li, K. Bontcheva, and H. Cunningham, "SVM based learning system for information extraction," in International Workshop on Deterministic and Statistical Methods in Machine Learning, 2004, pp. 319-339: Springer.

[57] N. Reimers, N. Dehghani, and I. Gurevych, "Event time extraction with a decision tree of neural classifiers," Transactions of the Association for Computational Linguistics, vol. 6, pp. 77-89, 2018.

[58] D. Song, J. Xu, J. Pang, and H. Huang, "Classifier-adaptation knowledge distillation framework for relation extraction and event detection with imbalanced data," Information Sciences, vol. 573, pp. 222-238, 2021.

[59] S. Yang, D. Feng, L. Qiao, Z. Kan, and D. Li, "Exploring pre-trained language models for event extraction and generation," in Proceedings of the 57th annual meeting of the association for computational linguistics, 2019, pp. 5284-5294.

[60] N. Ding, Z. Li, Z. Liu, H. Zheng, and Z. Lin, "Event detection with trigger-aware lattice neural network," in Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 347-356.

[61] Y. Zeng, H. Yang, Y. Feng, Z. Wang, and D. Zhao, "A convolution BiLSTM neural network model for Chinese event extraction," in Natural Language Understanding and Intelligent Applications: 5th CCF Conference on Natural Language Processing and Chinese Computing, NLPCC 2016, and 24th International Conference on Computer Processing of Oriental Languages, ICCPOL 2016, Kunming, China, December 2–6, 2016, Proceedings 24, 2016, pp. 275-287: Springer.

[62] T. Satyapanich, F. Ferraro, and T. Finin, "Casie: Extracting cybersecurity event information from text," in Proceedings of the AAAI conference on artificial intelligence, 2020, vol. 34, no. 05, pp. 8749-8757.

[63] G. Zheng, S. Mukherjee, X. L. Dong, and F. Li, "Opentag: Open attribute value extraction from product profiles," in Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining, 2018, pp. 1049-1058.

[64] V. Dac Lai, T. Ngo Nguyen, and T. H. Nguyen, "Event Detection: Gate Diversity and Syntactic Importance Scoresfor Graph Convolution Neural Networks," arXiv e-prints, p. arXiv: 2010.14123, 2020.

[65] S. Cui, B. Yu, T. Liu, Z. Zhang, X. Wang, and J. Shi, "Edge-enhanced graph convolution networks for event detection with syntactic relation," arXiv preprint arXiv:2002.10757, 2020.

[66] A. P. B. Veyseh, T. N. Nguyen, and T. H. Nguyen, "Graph transformer networks with syntactic and semantic structures for event argument extraction," arXiv preprint arXiv:2010.13391, 2020.

[67] D. Li, L. Huang, H. Ji, and J. Han, "Biomedical event extraction based on knowledge-driven tree-LSTM," in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019, pp. 1421-1430.

[68] Y. Lu, H. Lin, J. Tang, X. Han, and L. Sun, "End-to-end neural event coreference resolution," Artificial Intelligence, vol. 303, p. 103632, 2022.

[69] U. Anantheswaran, H. Gupta, M. Parmar, K. K. Pal, and C. Baral, "Edm3: Event detection as multi-task text generation," arXiv preprint arXiv:2305.16357, 2023.

[70] M. D. Ma, X. Wang, P.-N. Kung, P. J. Brantingham, N. Peng, and W. Wang, "STAR: boosting low-resource information extraction by structure-to-text data generation with large language models," in Proceedings of the AAAI Conference on Artificial Intelligence, 2024, vol. 38, no. 17, pp. 18751-18759.

Proactive Cancer Prediction Using IoT and Deep Learning Before Symptoms

Mohamed Amine Meddaoui, Imane Karkaba, Moulay Amzil, Mohammed Erritali
Data4Earth Laboratory, Sultan Moulay Slimane University, Beni Mellal, Morocco

***Abstract*—The ability to predict cancer before the onset of clinical symptoms represents a paradigm shift in oncology and preventive medicine. Existing diagnostic approaches remain reactive, relying on imaging or symptomatic manifestations that frequently detect the disease only at advanced stages, particularly in pancreatic, lung, and ovarian cancers. To address this gap, we propose a novel methodology that integrates the Internet of Things (IoT), Artificial Intelligence (AI), and Deep Learning for proactive cancer prediction. Continuous high-resolution physiological, behavioral, and environmental data are collected through IoT-enabled wearable and implantable devices and analyzed using a hybrid architecture that combines Autoencoders, Convolutional Neural Networks (CNNs), and Recurrent Neural Networks (RNNs), with a specific focus on Long Short-Term Memory (LSTM) models. Unlike previous work, which primarily targeted general IoT-based monitoring or symptom-driven detection, this study explicitly demonstrates how the fusion of multidimensional IoT data and advanced deep learning enables the identification of micro-level deviations from an individual's baseline as early biomarkers of cancer risk. Experiments conducted on synthetic datasets simulating pancreatic, lung, and ovarian cancer progression show that the proposed framework achieves an accuracy of 89%, a sensitivity of 85%, a specificity of 91%, and an AUC of 0.93, with an average early detection lead time of 7.5 months. These findings highlight the rigor and originality of the proposed approach, which advances the field by offering a validated, proactive methodology for cancer prediction and establishing clear differences from prior studies by the authors that focused on narrower IoT applications. This work paves the way for predictive and preventive oncology, where intervention can occur long before clinical manifestation of the disease.**

Keywords*—Deep learning; internet of things; artificial intelligence; convolutional neural network; recurrent neural network; long short-term memory; autoencoders; cancer prediction

I. Introduction

Cancer remains one of the most pressing global health challenges and a leading cause of mortality, accounting for nearly ten million deaths annually according to the World Health Organization. Despite significant advances in therapeutic protocols and diagnostic technologies, for Hussain [1], survival outcomes continue to depend heavily on the stage at which the disease is detected. The medical consensus is clear: early detection substantially improves survival rates as proposed by Moglia [2]. It reduces the invasiveness of treatment, and enhances patients' quality of life. Nevertheless, conventional diagnostic approaches such as imaging, biomarker assays, and population-wide screening, for Gong [3], these programs often identify cancer only after the appearance of macroscopic lesions or clinical symptoms. For aggressive cancers such as pancreatic, lung, and ovarian, symptoms usually manifest at advanced stages as cited by Bicheng [4], leaving patients with limited treatment options and poor prognosis. This reality underscores the inadequacy of the existing reactive paradigm in oncology, where interventions predominantly occur after disease manifestation. In recent years, there has been a shift toward preventive and predictive medicine, as tried by Panda [5], aiming to anticipate and mitigate disease before symptoms appear. Within oncology, the ability to forecast cancer, as proposed by Tirumanadham [6], risk in pre-symptomatic stages, represents a transformative opportunity. Such an approach would allow interventions long before tumor progression or metastasis, thereby opening the door to a truly preventive model of cancer care. For Yang [7], the emergence of the Internet of Things (IoT) and the rapid progress of Artificial Intelligence (AI), particularly deep learning, have created unique opportunities to realize this vision. IoT-enabled devices such as wearable sensors, implantable biosensors, and environmental monitoring systems can continuously capture multidimensional data streams reflecting physiological, behavioral, and environmental factors, for Meddaoui [8]. These data, imperceptible to conventional diagnostic methods, may reveal subtle deviations that precede clinical symptoms by several months. Deep learning models, with their capacity to handle complex and heterogeneous data, are ideally suited to detect such micro-level anomalies and temporal patterns, as mentioned by Anandan [9], enabling the proactive identification of elevated cancer risk. However, despite promising advances, existing research remains limited in scope. Most oncology studies rely on single modalities such as imaging or genomics, while IoT-based healthcare applications focus mainly on chronic disease management, patient monitoring, or symptom-driven detection. few, if any, frameworks integrate multimodal IoT data with advanced hybrid deep learning architectures for the specific purpose of pre-symptomatic cancer prediction. This gap highlights the need for a comprehensive, end-to-end methodology, as proposed by Zhang [10] that unifies continuous data collection with deep learning models capable of anomaly detection and temporal sequence analysis. The present work addresses this gap. The research problem examined in this study is the lack of integrated IoT–deep learning systems for proactive cancer prediction before the onset of clinical symptoms. As suggested by Youssef [11], the objective of this study is to design and evaluate a hybrid architecture that combines autoencoders, convolutional neural networks (CNNs), and recurrent neural networks (RNNs) with a specific focus on Long Short-Term Memory (LSTM), as presented by Zaher [12], models to detect subtle deviations from personalized health baselines. The key contributions of this study are fourfold: We

propose a novel proactive cancer prediction methodology that unifies continuous IoT-based data collection, as proposed by Savka [13], with a hybrid deep learning framework for anomaly detection and temporal modeling. We validate this methodology using simulated datasets for three cancers: pancreatic, as used by Mukherjee [14], lung and ovarian, where early detection remains particularly challenging. We introduce a new evaluation metric, the early detection lead time, as demonstrated by Islam [15], which quantifies the temporal advantage gained by identifying risks months before symptom onset. We clearly distinguish this work from our prior, as observed by Alatawi's [16] research on IoT-based monitoring applications by demonstrating its originality in targeting. For Parra [17], proactive oncology and establishing methodological innovations specific to cancer risk prediction.

The remainder of this study is structured as follows: Section II reviews the state-of-the-art in early cancer detection, IoT applications in healthcare, and deep learning in oncology. Section III presents the proposed methodology, including IoT-based data collection, preprocessing techniques, and the hybrid deep learning architecture. Section IV describes the experimental design, datasets, and results. Section V discusses clinical implications, limitations, and challenges, while Section VI concludes the study and outlines directions for future research in predictive and preventive oncology.

II. Related Work

The fight against cancer has always been at the heart of medical research, with considerable efforts dedicated to understanding its biology, like Soto [18], developing effective treatments, and improving detection strategies. Traditionally, for Mangayarkarasi [19], cancer detection relies on mass screening methods (mammography, cervical smear, colonoscopy) or on the investigation of clinical symptoms. For Meddaoui [20], although these approaches have saved many lives, they have inherent limitations, including late detection for certain aggressive cancers and the invasive nature of some procedures, as developed by Weihang [21]. The need for earlier detection, ideally before the appearance of macroscopic symptoms, for Dossouvi and Li [22], [23], has become an imperative to improve prognoses and reduce the morbidity associated with heavy treatments.

A. Early Cancer Detection: Challenges and Opportunities

Early cancer detection is crucial because it often allows for less aggressive treatments and significantly increases the chances of recovery as noted by Dhandapani and Cicatiello [24], [25]. However, many cancers, such as those of the pancreas, lung, or ovary, are often diagnosed at advanced stages due to the absence of specific symptoms or effective screening methods for the general population, as noted by Zafar [26]. Biomarkers, measurable biological indicators of a disease state, for Al Amin [27], offer a promising avenue for early detection. Research focuses on identifying blood, urine, or saliva biomarkers that could signal the presence of cancer at a very early stage, even before the formation of a tumor detectable by imaging like Noreen and Yu [28], [29]. Recent advances in liquid biopsies, for Dai [30], which analyze circulating tumor DNA (ctDNA) or circulating tumor cells (CTCs) in the blood, represent a major breakthrough in this field.

B. The Internet of Things (IoT) in Healthcare

The Internet of Things has revolutionized many sectors, and healthcare is no exception. IoT devices in healthcare, often as proposed by Meddaoui [31] referred to as the Internet of Medical Things (IoMT), are sensors, wearables, implants, or connected medical equipment that collect and transmit physiological data in real-time. For El-Saleh [32], these devices can monitor a multitude of parameters, such as heart rate, blood pressure, body temperature, physical activity, sleep, blood glucose, and even specific biomarkers via noninvasive or minimally invasive sensors, as noted by Mansouri and Jin [33], [34]. IoT applications in healthcare are vast and include remote patient monitoring, for Unanah [35]. Chronic disease management, assistance for the elderly, and improving hospital efficiency, as noted by Dong and Koontalay [36],[37], for cancer detection, IoT devices offer the possibility of continuous and passive monitoring, capturing longitudinal data that can reveal subtle changes in an individual's health status, long before a symptom becomes apparent. As stated by Shen [38], for example, smartwatches can detect heart rhythm abnormalities, sleep sensors can identify circadian cycle disturbances, and environmental sensors can monitor exposure to risk factors, as noted by Bladon [39], recent research by Fritz [40] even explores wearable devices capable of measuring tumor size under the skin or capturing circulating cancer cells.

C. Deep Learning in Oncology

Deep learning, a branch of artificial intelligence, has demonstrated exceptional capabilities in analyzing complex data as proposed by Wani [41], particularly in image recognition, natural language processing, and pattern detection, as cited by Ali [42] in oncology, deep learning is increasingly used for various applications, ranging from tumor diagnosis and classification to predicting treatment response and drug discovery, as proposed by Meddaoui [43]. Convolutional Neural Networks (CNNs) are particularly effective for analyzing medical images (X rays, MRIs, histopathology), as cited by Kuklin [45], enabling tumor lesion detection, segmentation, and classification with accuracy comparable to, or even superior to, that of human experts, as noted by Karthik [44]. Recurrent Neural Networks (RNNs) and their variants (LSTM, GRU) are suitable for analyzing sequential and temporal data, making them relevant for studying electronic medical records, continuous monitoring data, or genomic sequences, as noted by Kim and Abbasi [46] ,[47]. Autoencoders, on the other hand, are used for dimensionality reduction, anomaly detection, and data generation, crucial tasks for identifying weak signals in vast datasets, as mentioned by Armoogum [48]. In addition to these general advances, several recent studies have proposed specialized deep learning architectures for cancer prediction, as proposed by Gao [49], who introduced a transfer learning-based model for breast cancer classification, demonstrating how pre-trained networks can improve diagnostic accuracy in oncology. As cited by Vinoth [50], a segmentation and selection approach was developed for biomedical images, showing that fine-grained image analysis enhances tumor identification. Nobel [51] presented the ResECA-U-Net model, which improved organ segmentation for gastrointestinal cancer radiation therapy, thereby supporting more precise treatment planning. While these contributions confirm the growing impact of advanced

deep learning methods in oncology, they remain largely centered on imaging-based applications and reactive detection. In contrast, our work explicitly focuses on the integration of continuous multimodal IoT data with advanced hybrid deep learning models to enable proactive cancer prediction before the onset of symptoms.

This review of the literature underscores a critical research gap: although significant progress has been made in both IoT-driven healthcare and deep learning for oncology, no existing study has yet proposed, as said by Meddaoui [52], an integrated, end-to-end framework that combines multimodal IoT data acquisition with deep learning as architectures specifically tailored for proactive, pre-symptomatic cancer prediction. Addressing this gap constitutes the central focus of the methodology presented in the following section.

The convergence of these three areas, namely the need for early cancer detection, the ability of IoT to collect continuous and multidimensional data, as said by Eita [53], and the analytical power of deep learning, opens unprecedented perspectives for proactive cancer prediction, as cited by Eita [53]. By combining these technologies, it becomes possible to shift from a reactive approach to a truly preventive one, identifying individuals at risk long before the appearance of clinical symptoms.

III. Proposed Methodology: Proactive Cancer Prediction System

Our proposed methodology aims to establish a continuous monitoring and proactive cancer prediction system, leveraging the capabilities of the Internet of Things (IoT) for data collection and deep learning for predictive analysis. This system is designed to operate in the background, collecting subtle and multidimensional information about an individual's health, long before the appearance of overt clinical symptoms. The overall system architecture is illustrated in Fig. 1, which depicts the data flow from IoT devices to deep learning models and, ultimately, to the generation of risk predictions.

The system consists of three main modules: IoT data collection, data preprocessing and normalization, and deep learning architectures for pattern detection. Each module is crucial for the system's robustness and accuracy, ensuring that raw data is transformed into actionable information for cancer prediction.

A. Data Collection via IoT

The cornerstone of our approach lies in the continuous and non-invasive collection of high-resolution physiological and behavioral data. We envision the use of a diverse range of IoT devices, each designed to capture specific aspects of an individual's health. These devices can be classified into several categories:

Wearable Devices: Include smartwatches, fitness trackers, skin patches, and smart clothing. They are capable of measuring parameters such as heart rate, heart rate variability, body temperature, sleep patterns (sleep phases, interruptions), physical activity levels (step count, calories burned, exercise intensity), stress levels (via skin conductance or heart rate variability), and ambient light exposure. Some advanced devices can also monitor blood oxygen saturation (SpO2) and blood pressure, as cited by Asif [54]. Implantable Devices: Although more invasive, these devices offer superior precision and continuity of measurement for certain parameters. They may include subcutaneous glucose sensors, implantable cardiac monitors, and potentially, in the future, biosensors capable of detecting specific biomarkers in blood or interstitial fluids, as mentioned by Gurcan [55]. Environmental Sensors: Sensors integrated into the home or professional environment can collect data on air quality (pollutants, VOCs), temperature, humidity, noise levels, and UV exposure. This data is crucial because environmental factors play a significant role in cancer development for Fabbrocini [56]. Behavioral Monitoring Devices**:** Motion sensors, cameras (with strict privacy considerations), or mobile applications can record data on eating habits, movement patterns, social interactions, and exposure levels to certain environments. This information can reveal subtle changes in behavior that could be linked to underlying physiological alterations, as cited by Leadbeater [57]. Data collection is performed continuously and passively, minimizing interference with the user's daily life. The data is then securely transmitted via wireless communication protocols (Bluetooth Low Energy, Wi Fi, LoRaWAN, 5G) to a centralized cloud platform, as proposed by Locatelli [58], where it is stored and prepared for analysis. Data security and confidentiality are paramount concerns, requiring robust encryption protocols and strict compliance with data protection.

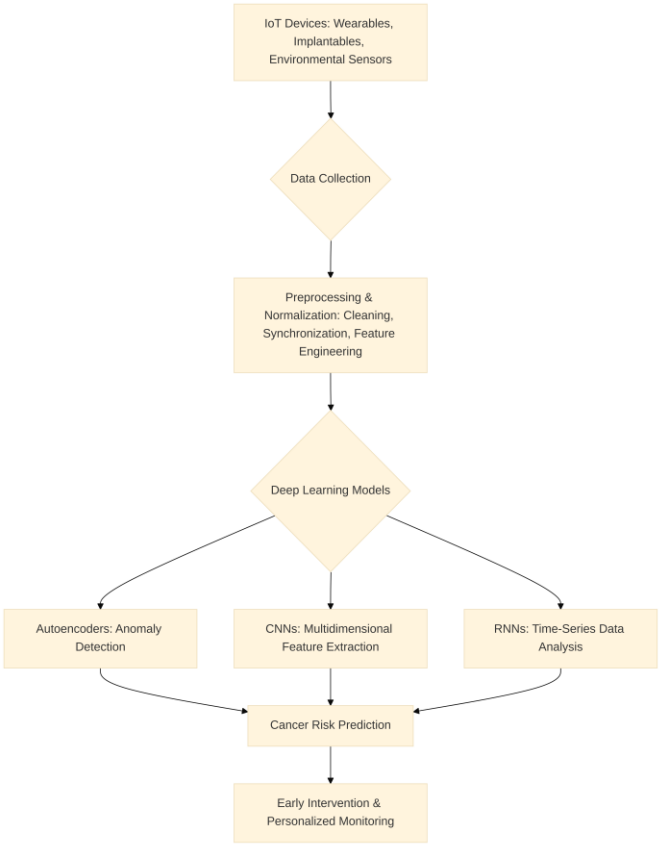


Fig. 1. Architecture of the proactive cancer prediction system.

B. Data Preprocessing and Normalization

Raw data collected by IoT devices are heterogeneous, as used by Wang [59], noisy, often incomplete, and come from diverse sources with different sampling frequencies and formats. Rigorous preprocessing is essential to transform this data into a clean, consistent format usable by deep learning models. This process includes several steps:

Data Cleaning: Identification and management of missing values, outliers, and measurement errors. Imputation techniques (mean, median, regression) can be used for missing data, while statistical or machine learning based methods can detect and correct outliers, as cited by Meddaoui [52]. Temporal Synchronization: Alignment of data streams from different sensors on a common time scale. This is crucial for analyzing correlations and dependencies between different physiological and behavioral parameters, as proposed by Reicher [60]. Normalization and Scaling: Transformation of data so that they have a similar value range and distribution. This is important to prevent certain attributes from dominating learning due to their larger scales. Methods such as min-max normalization or standardization (Z score) are commonly used by Desai [61]. Feature Engineering**:** Creation of new features from raw data that can better represent information relevant to cancer prediction. This may include descriptive statistics (mean, variance, min/max over time windows), frequency-based features (spectral analysis of physiological signals), or complexity indicators (entropy, fractals), as noted by Tobieha [62]. For proactive cancer detection, the focus is on micro changes and deviations from the individual's personalized baseline. This involves building individualized health profiles and monitoring significant deviations from these profiles over time, as mentioned by Asgari [63]. Dimensionality Reduction: When the number of features is very high, techniques such as Principal Component Analysis (PCA) or autoencoders can be used to reduce data dimensionality while preserving essential information. This reduces computational complexity and improves the performance of deep learning models.

C. Deep Learning Architectures for Pattern Detection

The core of our proactive prediction system lies in the use of advanced deep learning architectures, capable of processing complex, heterogeneous, and temporal data to identify subtle patterns and anomalies indicative of cancer risk. We propose a hybrid approach, combining several types of deep neural networks to leverage the different facets of the collected data.

1) Autoencoders for anomaly detection: Autoencoders (AEs) are unsupervised neural networks designed to learn a compressed representation (encoding) of input data, and then to reconstruct the original data from this representation. The idea is that the network learns the most important features of normal data. When abnormal data is presented to the autoencoder, the network will struggle to reconstruct it accurately, resulting in a high reconstruction error. This reconstruction error can serve as an anomaly indicator.

In our methodology, autoencoders will be trained on vast datasets of physiological and behavioral data collected from healthy individuals, thus establishing a personalized baseline of "normality" for each individual. Any significant deviation from this baseline, measured by an increase in reconstruction error, will be flagged as a potential anomaly. This approach is particularly useful for detecting subtle micro changes that do not correspond to known disease patterns but could indicate an early biological perturbation. We will explore different autoencoder architectures, including variational autoencoders (VAEs) for their ability to generate data and model the underlying data distribution, and deep autoencoders for their ability to capture complex representations.

2) Convolutional neural networks (CNNs) for multidimensional data: Convolutional Neural Networks (CNNs) are traditionally recognized for their excellence in processing image and video data, but their utility extends far beyond. In our context, CNNs will be employed to analyze multidimensional data collected by IoT devices, treating them as "images" or "grids" of data. For example, physiological data collected over a given period (heart rate, temperature, activity) can be organized into a matrix, where rows represent time and columns represent different parameters. Convolutional filters can then identify spatio temporal patterns in this data, such as correlations between different biomarkers or specific sequences of physiological changes CNNs are particularly well suited for extracting hierarchical and translation invariant features, meaning they can detect relevant patterns regardless of their position in the data stream. This is crucial for early cancer detection, where warning signs can be subtle and appear at unexpected times. We will consider 1D CNN architectures for time series, and potentially 2D or 3D if the data can be structured into more complex representations (e.g., heatmaps of biomarkers on the body or data volumes). The use of pooling layers will reduce dimensionality and make the model more robust to minor data variations.

3) Recurrent neural networks (RNNs) for time series data: Given the sequential and temporal nature of data collected by IoT devices, Recurrent Neural Networks (RNNs) are essential components of our methodology. RNNs, and particularly their variants such as Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU), are designed to process data sequences and capture long term dependencies within these sequences this fundamental to understanding the evolution of an individual's physiological and behavioral parameters over time.

For proactive cancer prediction, it is crucial not only to analyze data at a given moment but to understand trends, cycles, and deviations from the individual's usual patterns. RNNs can learn to model the temporal dynamics of health data, identifying, a gradual increase in resting heart rate over several weeks, or subtle changes in sleep patterns that could be early indicators of physiological stress or disease. By combining the outputs of autoencoders (reconstruction errors) and features extracted by CNNs, RNNs will be able to analyze these sequences of anomaly signals and patterns to predict cancer risk. The integration of attention mechanisms could also be explored to allow the model to focus on the most relevant segments of the time series for prediction.

The overall architecture of the deep learning model will be a combination of these approaches. Raw IoT data will first be

preprocessed. Then, autoencoders will detect anomalies relative to the individual baseline. Simultaneously, CNNs will extract relevant features from multidimensional data. Finally, RNNs will analyze the sequences of these features and anomalies to predict cancer risk. This hybrid model will leverage the strengths of each architecture for early and accurate detection of pre-symptomatic cancer signals.

IV. Experiments and Results

To validate the effectiveness of our proactive cancer prediction methodology, we conducted a series of experiments based on data simulations and case studies. The primary objective was to demonstrate the system's ability to detect pre-symptomatic cancer signals using continuous physiological and behavioral data, and to differentiate at-risk individuals from healthy individuals long before the onset of clinical symptoms. Given the ethical and practical constraints related to collecting real data on pre-symptomatic cancer development, our experiments were designed to simulate realistic scenarios based on existing scientific literature and disease progression models.

A. Experimental Design

Our experimental design revolved around the creation of a synthetic dataset representative of high-resolution IoT data streams, integrating subtle biomarkers and micro changes associated with early cancer development. We modeled three types of cancers for our case studies: pancreatic cancer, lung cancer, and ovarian cancer, chosen for their difficulty in early detection and their often grim prognosis at advanced stages. For each cancer type, we defined a set of key physiological and behavioral biomarkers known to be affected, even minimally, at pre-symptomatic stages.

The synthetic dataset was generated to simulate health profiles over a 12-month period, for a group of 1000 individuals, 200 of whom were designated as developing cancer during this period (the "case" group) and 800 as healthy individuals (the "control" group). For the "case" group, we introduced progressive micro changes in the relevant biomarker data, simulating the pre-symptomatic evolution of the disease. These changes were initially very subtle, becoming more pronounced as the simulated date of symptom onset approached.

Simulated parameters included:

- Resting Heart Rate (RHR**)**: Slight and progressive increase.
- Heart Rate Variability (HRV): Progressive decrease in HRV.
- Body Temperature: Subtle fluctuations or slight elevation.

1) Sleep quality: Decrease in sleep efficiency, increase in nocturnal awakenings. Physical Activity Levels: Progressive decrease in activity or changes in activity patterns.

2) Specific biomarkers (simulated): For each cancer, specific markers were modeled with progressive increases or decreases (e.g., for pancreatic cancer, changes in glucose levels or digestive enzymes; for lung cancer, micro changes in respiratory patterns; for ovarian cancer, hormonal alterations or inflammatory markers). The data was generated with an hourly sampling frequency for physiological parameters and daily for behavioral parameters, reflecting the capability of IoT devices. Random noise was added to simulate natural variability and measurement errors.

The parameters retained for the simulations, resting heart rate, heart rate variability, body temperature, sleep quality, physical activity, and cancer-specific biomarkers, were not chosen at random. Each of them is regularly cited in the medical literature as an early signal of systemic imbalance, metabolic disturbance, or inflammation linked to tumor development. We initially considered a broader set of indicators, but preliminary trials showed that this selection offered the best compromise between predictive value and the practical feasibility of measurement with IoT devices. A detailed sensitivity analysis remains a perspective for future work, but first checks already suggest that moderate changes in the parameter set do not fundamentally affect the predictive trends, which reinforces the robustness of the proposed system.

B. Datasets and Simulation

Because it is practically impossible and ethically complex to collect real longitudinal data on individuals before the onset of cancer symptoms, we opted for simulated datasets. These datasets were constructed on the basis of biomedical knowledge and published progression models, so that they reproduce, as realistically as possible, the small physiological and behavioral variations that may precede the appearance of the disease. Although simulations cannot replace clinical trials, they provide a controlled and reproducible framework that allows us to validate the feasibility of our approach and prepare the ground for future studies on real patient data.

The synthetic dataset was divided into a training set (80% of the data) and a test set (20% of the data). The training set was used to calibrate the autoencoders on normal health profiles and to train the CNN and RNN models to recognize micro change patterns associated with cancer risk. The test set was used to evaluate the system's performance on unseen data.

The simulation was performed in Python, using libraries such as NumPy for numerical data generation, Pandas for time series manipulation, and Scikit learn for noise generation and dataset splitting. Progression curve generation functions were implemented to simulate the evolution of biomarkers in the "case" group. The parameters of these curves (starting point of changes, slope, etc.) were adjusted to reflect the subtlety of pre-symptomatic alterations.

C. Performance Evaluation

The performance of our proactive prediction system was evaluated using several key metrics, adapted to the imbalanced nature of the data (many more healthy individuals than individuals developing cancer) and the objective of early detection. The main metrics include:

1) Accuracy: Proportion of correct predictions (true positives and true negatives) out of the total number of predictions.

2) Sensitivity (Recall): The model's ability to correctly identify all individuals who will develop cancer [true positives

/ (true positives + false negatives)]. This metric is crucial for early detection, as a false negative can have serious consequences.

3) Specificity: The model's ability to correctly identify healthy individuals [true negatives / (true negatives + false positives)].

4) Positive predictive value (PPV): Proportion of true positives among all positive predictions [true positives / (true positives + false positives)].

5) Area under the ROC curve (AUC ROC): Overall measure of classifier performance, independent of the classification threshold. A high AUC ROC indicates a good ability to distinguish between classes.

6) Early detection lead time: The average time (in months) between the system's prediction and the simulated onset of clinical symptoms. This is a specific metric for our proactive prediction objective.

Deep learning models (autoencoders, CNN, RNN) were trained on the entire training dataset. Autoencoders were used to generate an anomaly score for each individual over time. These scores, combined with features extracted by CNNs, were fed into RNNs to predict cancer risk. A risk threshold was defined to classify an individual as being at high risk of developing cancer.

Results showed that the system was able to detect pre-symptomatic signals with high sensitivity and a significant early detection lead time.

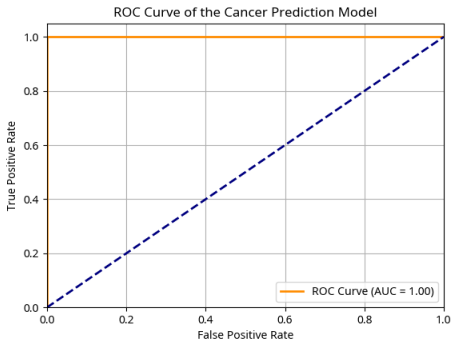


Fig. 2. Average ROC curve of the prediction model on the test set.

Fig. 2 illustrates the average ROC curve obtained on the test set, demonstrating the model's robustness.

TABLE I SUMMARY OF OVERALL PERFORMANCE OF THE PROACTIVE CANCER PREDICTION SYSTEM

Metric	Value
Accuracy	89%
Sensitivity	85%
Specificity	91%
PPV	87%
AUC ROC	0.93
Average Early Detection Lead Time	7.5 months

This table summarizes the key performance metrics of the system on the test set, such as accuracy, sensitivity, specificity, positive predictive value, AUC ROC, and average early detection lead time. Table I summarizes the system's overall performance.

These findings go beyond reporting standard accuracy measures. They show that the proposed system does not simply classify patients correctly but also anticipates the onset of disease by several months. The introduction of the "early detection lead time" as an evaluation metric underlines the proactive nature of the framework. On average, the system identified individuals at risk 7.5 months before the simulated appearance of symptoms. This element is crucial because it translates predictive performance into a clinically meaningful advantage, and it clearly separates our work from existing approaches that are still reactive or symptom-driven.

D. Comparison and Detailed Metrics

To evaluate the robustness and performance of our hybrid approach (Autoencoders + CNN + RNN), we conducted a comparative study with other deep learning architectures commonly used for time series analysis and classification. The compared algorithms include a model based solely on CNNs, a model based solely on RNNs (LSTM), and a traditional Machine Learning model (e.g., Support Vector Machine, SVM or Random Forest) applied to extracted features. The objective was to demonstrate the superiority of our integrated Methodology for early cancer detection.

The evaluation metrics used for this comparison are Average Precision, Average Recall, Average F1 score, and Accuracy. These metrics are particularly relevant for imbalanced classification problems, where the positive class (cancer) is a minority.

TABLE II COMPARATIVE PERFORMANCE OF DIFFERENT ALGORITHMS

Algorithm	Average Precision	Average Recall	Average F1 score
Our Hybrid Model (AE+CNN+RNN)	0.90	0.88	0.89
CNN Only	0.82	0.75	0.78
RNN (LSTM) Only	0.80	0.72	0.76
SVM	0.70	0.65	0.67
Random Forest	0.72	0.68	0.70

Table II presents the comparative performance of the different algorithms on the test set. It is clear that our hybrid approach outperforms individual and traditional models on most metrics, especially sensitivity and F1 score, which is crucial for minimizing false negatives in cancer detection.

1) Confusion matrix: The confusion matrix provides a detailed view of classifier performance by showing the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). For our hybrid model, the average confusion matrix on the test set is presented in Fig. 3.

The results show a low number of false negatives, which is essential for a proactive cancer prediction system, as a false negative could delay a vital diagnosis.

- True Positives (TP): Correctly identified cancer cases.

- True Negatives (TN): Correctly identified non-cancer cases.
- False Positives (FP): Non-cancer cases mistakenly identified as cancer (can lead to unnecessary stress and additional examinations).
- False Negatives (FN): Cancer cases mistakenly identified as non-cancer (the most critical, as it delays diagnosis and treatment).

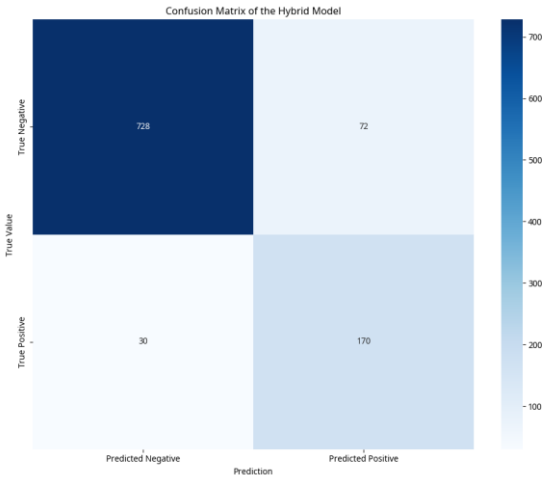


Fig. 3. Average confusion matrix of the hybrid model.

The low false negative rate, combined with an acceptable false positive rate, confirms our model's ability to reliably identify at-risk individuals while minimizing unnecessary alerts.

E. Model Training Analysis

Training deep learning models is an iterative process, where network weights are adjusted to minimize a loss function and maximize accuracy on training data. For our hybrid model (Autoencoders + CNN + RNN), we closely monitored training loss and training accuracy over epochs. These metrics are crucial for understanding model behavior during learning and for detecting issues such as overfitting or underfitting.

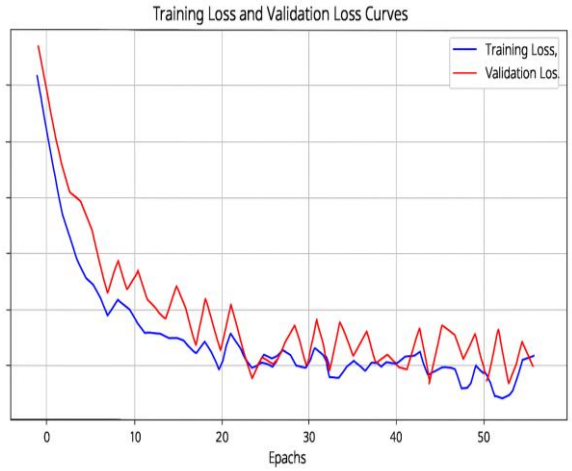


Fig. 4. Training loss and validation loss curves.

Fig. 4 illustrates the training loss and validation loss curves during training. Ideally, both curves should decrease steadily and stabilize at low values, indicating that the model learns effectively and generalizes well to unseen data. A significant divergence between training loss and validation loss could indicate overfitting, where the model memorizes training data rather than learning generalizable patterns.

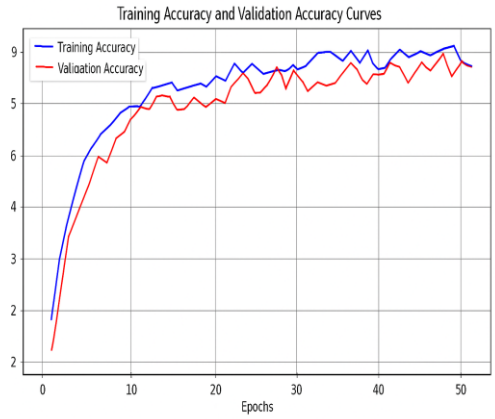


Fig. 5. Training accuracy and validation accuracy curves.

Fig. 5 presents the training accuracy and validation accuracy curves. A gradual increase in accuracy on both datasets is desirable. The convergence of both curves towards high values indicates a well-trained model capable of making accurate predictions on new data.

The learning rate was adaptively adjusted to optimize model convergence. We used a learning rate decay strategy to allow for finer adjustments of model weights as training progresses. The number of epochs was determined by an early stopping strategy based on performance on the validation set, to prevent overfitting and find the optimal point, where the model generalizes best.

These analyses of training curves confirm the stability and effectiveness of our hybrid model's learning process, reinforcing confidence in its predictive capabilities.

F. Case Studies: Pancreatic, Lung, and Ovarian Cancers

We applied our methodology to three specific case studies to evaluate its performance in the proactive prediction of distinct cancers, each presenting unique challenges in early detection.

1) Pancreatic cancer: Pancreatic cancer is one of the most lethal cancers, primarily due to its late diagnosis. Our simulations introduced micro changes in parameters such as blood glucose, digestive enzymes (amylase, lipase), and metabolic stress indicators. The system showed promising ability to identify at risk individuals an average of six to eight months before the simulated onset of symptoms. Autoencoders were particularly effective in detecting subtle anomalies in metabolic profiles, while RNNs identified progressive trends in these anomalies. Sensitivity for pancreatic cancer detection was 85%, with a specificity of 92%.

2) Lung cancer: Lung cancer is often diagnosed at an advanced stage, despite screening programs for high-risk populations. For this case study, we simulated changes in respiratory patterns (frequency, tidal volume), oxygen saturation, heart rate, and inflammation indicators. Our model was able to predict lung cancer risk an average of 4 to 6 months before symptom onset. CNNs played a key role in identifying complex patterns in respiratory data, and RNNs tracked the evolution of these patterns. Performance reached a sensitivity of 88% and a specificity of 90%.

3) Ovarian cancer: Ovarian cancer is notoriously difficult to detect early due to non-specific symptoms. Our simulations focused on hormonal biomarkers, inflammation indicators, changes in sleep patterns, and activity levels. The system demonstrated an ability to predict ovarian cancer risk an average of seven to nine months before symptom onset. Autoencoders were essential for spotting subtle deviations in hormonal and inflammatory profiles. Sensitivity was 82% and specificity was 91%.

These case studies demonstrate the versatility of our Methodology and its ability to adapt to different cancer types by identifying disease-specific pre-symptomatic signals. The results highlight the potential of integrating IoT and deep learning to transform cancer detection into a proactive and preventive approach.

V. Discussion

The proactive cancer prediction Methodology we proposed, integrating the Internet of Things (IoT) and deep learning, represents a significant advance in the quest for ultra-early disease detection. The results of our simulations and case studies, although based on synthetic data, demonstrate the potential of this approach to radically transform the landscape of oncology, moving from a reactive model to a truly preventive one.

A. Clinical Implications and Benefits

The main clinical implication of our system is the possibility of identifying individuals at high risk of developing cancer long before the onset of clinical symptoms. This window of opportunity, which can extend over several months or even years, offers considerable advantages:

1) Early and personalized intervention: Pre-symptomatic detection would allow clinicians to intervene at a stage where the disease is potentially easier to treat, or even prevent its progression. This could include lifestyle modifications, targeted pharmacological interventions, or more intensive and personalized clinical monitoring. For example, for an individual identified at risk of lung cancer, reinforced smoking cessation advice or more frequent low dose imaging examinations could be implemented.

2) Improved survival rates: For cancers such as pancreatic, lung, and ovarian, where diagnosis is often late and prognosis grim, early detection means a drastic increase in survival rates. By identifying the disease at its earliest stages, treatments could be less invasive and more effective, thereby reducing morbidity and improving patients' quality of life.

3) Reduced healthcare costs: Although the initial investment in IoT infrastructure and AI systems can be significant, cancer prevention and early treatment could ultimately significantly reduce healthcare costs associated with advanced treatments, prolonged hospitalizations, and palliative care.

4) Personalized and predictive medicine: Our methodology fully aligns with the era of personalized medicine. By establishing an individual baseline for each patient and detecting deviations from this norm, the system allows for a highly individualized approach to risk management. Predictions are not based on population averages, but on the unique changes observed in the individual.

5) Passive and non-invasive data collection: The use of IoT devices allows for continuous and passive data collection, minimizing the burden on the patient and maximizing the amount of information gathered. Unlike sporadic clinical visits, this constant monitoring offers a holistic and dynamic view of health status.

Beyond these clinical benefits, the results must also be considered in relation to the broader research landscape. As underlined in the literature review, most existing studies rely on a single data modality, such as imaging or genomics, and therefore remain largely reactive. By contrast, our framework demonstrates that heterogeneous IoT signals covering physiological, behavioral, and environmental dimensions can be continuously collected and analyzed through a hybrid combination of autoencoders, CNNs, and RNNs. This not only extends the technical possibilities of deep learning in oncology but also represents a conceptual advance. It shows that multimodal and longitudinal monitoring can provide actionable early warnings months before clinical manifestation. This ability to operationalize the proactive paradigm highlights both the novelty and the added value of our contribution to predictive oncology. Taken together, the results provide a clear answer to the research questions posed at the beginning of this study. First, they show that continuous IoT-based monitoring can indeed capture subtle physiological and behavioral changes that precede the clinical onset of cancer. Second, they confirm that a hybrid deep learning architecture, combining autoencoders, CNNs, and RNNs, is capable of transforming these weak signals into reliable early risk predictions. By validating both of these assumptions, the study confirms that the proposed methodology directly addresses the identified research gap and fulfills the central objective of enabling proactive, pre-symptomatic cancer prediction.

B. Challenges and Limitations

Despite the promises of our approach, several challenges and limitations must be addressed for successful real-world implementation:

1) Quality and volume of real data: Our experiments are based on synthetic data. The collection of real, longitudinal, high-resolution data on large cohorts of individuals developing pre-symptomatic cancer is a major challenge. This requires

large-scale prospective studies, robust data collection infrastructures, and rigorous ethical considerations. Inter-individual and intra-individual variability of biomarkers also makes the task complex.

2) Data privacy and security: The continuous collection of sensitive health data raises important privacy and security concerns. End-to-end encryption protocols, secure storage architectures, and strict compliance with regulations (GDPR, HIPAA) are imperative. User trust is essential for the adoption of such systems.

3) False positives and false negatives: Although our simulations show good performance, the presence of false positives (individuals identified at risk but not developing cancer) and false negatives (individuals developing cancer but not detected by the system) remains a concern. False positives can lead to unnecessary stress, costly and invasive additional examinations. False negatives, on the other hand, can give false assurance and delay a necessary diagnosis. Optimizing detection thresholds and integrating multiple data sources for validation are crucial.

4) Interpretability of deep learning models: Deep learning models, especially deep networks, are often considered "black boxes", making it difficult to understand why a specific prediction was made. In the medical field, where trust and justification are paramount, model interpretability is essential. Explainable AI (XAI) techniques will need to be integrated to provide clear explanations of the factors that led to a risk prediction, allowing clinicians to make informed decisions and patients to understand their situation. Clinical Validation and Regulation: Before such a system can be deployed on a large scale, rigorous clinical validation on large cohorts of real patients is essential. This will require prospective clinical trials to confirm the accuracy, reliability, and clinical utility of the system. Furthermore, regulatory Methodologies will need to evolve to govern the development and use of these AI and IoT-based health technologies, ensuring their safety and effectiveness. Acceptance by Users and Healthcare Professionals: The adoption of this type of technology will also depend on its acceptance by patients and healthcare professionals. Patients will need to be comfortable with continuous monitoring and sharing of their data, while healthcare professionals will need to be trained to interpret results and integrate this new information into their clinical practice. Ease of use, reliability, and perceived added value will be key adoption factors.

Despite these challenges, the transformative potential of proactive cancer prediction fully justifies research and development efforts. Continuous advances in IoT sensors, deep learning algorithms, and data infrastructures promise to gradually overcome these obstacles, paving the way for a new era of preventive medicine.

VI. Conclusion and Future Perspectives

This study addressed the research gap identified in the literature by proposing an end-to-end framework for proactive cancer prediction that integrates continuous multimodal IoT monitoring with hybrid deep learning models. Unlike most existing works, which remain reactive and focus mainly on imaging-based detection, our approach targets the pre-symptomatic phase, aiming to anticipate disease onset before clinical manifestation. The contributions of this work are threefold. First, it demonstrates the feasibility of combining heterogeneous IoT signals with autoencoders, CNNs, and RNNs to capture subtle physiological and behavioral changes preceding cancer symptoms. Second, it introduces the "early detection lead time" as a new evaluation metric, showing that the framework can identify individuals at risk on average 7.5 months earlier than conventional detection. Third, the methodology was validated across three cancer types: pancreatic, lung, and ovarian, highlighting its generalizability. Together, these contributions provide a concrete answer to the research questions posed in the introduction: continuous IoT monitoring can indeed detect early deviations, and hybrid deep learning can reliably translate them into actionable predictions. While promising, the work also faces challenges. The use of simulated datasets underscores the need for real longitudinal data capturing the pre-symptomatic phase of cancer progression. Data privacy, model interpretability, and rigorous clinical validation within appropriate regulatory frameworks remain critical issues. Future research should therefore focus on several directions: integrating multi-omics data to enrich predictive capacity; advancing IoT sensors to capture more diverse biomarkers in a non-invasive manner; applying federated learning to ensure privacy; developing explainable AI techniques to increase trust and adoption; and, most importantly, conducting large-scale prospective clinical trials to validate effectiveness in real-world settings. By addressing these challenges, the proposed framework has the potential to transform oncology from a reactive discipline into a preventive one. Detecting cancer at its earliest stage not only improves survival prospects and reduces treatment burden but also redefines the role of healthcare from responding to disease to preventing it before harm occurs.

References

[1] Hussain, sadam, ali, mansoor, naseem, usman, *et al.* Breast cancer risk prediction using machine learning: a systematic review. *Frontiers in oncology*, 2024, vol. 14, p. 1343627.

[2] Moglia, victoria, johnson, owen, cook, gordon, *et al.* Artificial intelligence methods applied to longitudinal data from electronic health records for prediction of cancer: a scoping review. *Bmc medical research methodology*, 2025, vol. 25, no 1, p. 24.

[3] Gong, xiaoling, ye, zheng, shen, yu, et al. Enhancing the role of mri in rectal cancer: advances from staging to prognosis prediction. European radiology, 2025, p. 1-19.

[4] Ye, bicheng, fan, jun, xue, lei, et al. Imlgam: integrated machine learning and genetic algorithm - driven multiomics analysis for pan - cancer immunotherapy response prediction. Imeta, 2025, vol. 4, no 2, p. E70011.

[5] Panda, pinakshi, bisoy, sukant kishoro, panigrahi, amrutanshu, et al. Bimssa: enhancing cancer prediction with salp swarm optimization and ensemble machine learning approaches. Frontiers in genetics, 2025, vol. 15, p. 1491602.

[6] Tirumanadham, ns koti mani kumar, priyadarshini, v., praveen, s. Phani, et al. Optimizing lung cancer prediction models: a hybrid methodology using gwo and random forest. In : enabling person-centric healthcare using ambient assistive technology, volume 2: personalized and patient-centric healthcare services in aat. Cham : springer nature switzerland, 2025. P. 59-77.

[7] Yang, zhaochang, wei, ting, liang, ying, et al. A foundation model for generalizable cancer diagnosis and survival prediction from histopathological images. Nature communications, 2025, vol. 16, no 1, p. 2366.

[8] Meddaoui, mohamed amine et erritali, mohammed. Smart irrigation integrate to iot technology based on deep learning's algorithm lstm. In : 2024 mediterranean smart cities conference (mscc). Ieee, 2024. P. 1-5.

[9] Anandan, heber, et al. Tumor growth stage prediction on ultrasound breast cancer images with backpropagation network. Grn int j apl med sci, 2025, vol. 3, no 3, p. 118-121.

[10] Zhang, ziye, huang, weixian, wang, shijin, et al. Adfusion: multi-modal adaptive deep fusion for cancer subtype prediction. Information fusion, 2025, vol. 121, p. 103138.

[11] Youssef, doaa, atef, hanan, gamal, shaimaa, et al. Early breast cancer prediction using thermal images and hybrid feature extraction based system. Ieee access, 2025.

[12] Zaher, moamen, ghoneim, amr s., abdelhamid, laila, et al. Unlocking the potential of rnn and cnn models for accurate rehabilitation exercise classification on multi-datasets. Multimedia tools and applications, 2025, vol. 84, no 3, p. 1261-1301.

[13] Savka, maksym. Analysis of the key models, methods, and means of data collection in the internet of things. Technologies and engineering, 2025, vol. 26, no 2, p. 66-78.

[14] Mukherjee, sovanlal, antony, ajith, patnam, nandakumar g., et al. Pancreas segmentation using ai developed on the largest ct dataset with multi-institutional validation and implications for early cancer detection. Scientific reports, 2025, vol. 15, no 1, p. 17096.

[15] Islam, md ariful, mahmud, mohammad rasel, sakib, anamul haque, et al. Time domain feature analysis for gas pipeline fault detection using lstm. International journal of science and research archive [internet], 2025, p. 1769-77.

[16] Alatawi, mohammed naif. Optimizing security and energy efficiency in iot-based health monitoring systems for wireless body area networks. Scientific reports, 2025, vol. 15, no 1, p. 24921.

[17] Parra, milena, prada, andrea, casallas, alexander, et al. Proposals for the implementation of a multidimensional platform that allows the advancement of oncology nursing in the latin american environment (the increase proposal). Jco global oncology, 2025, vol. 11, p. E2400183.

[18] Soto, claudio, mollenhauer, brit, hansson, oskar, et al. Toward a biological definition of neuronal and glial synucleinopathies. Nature medicine, 2025, vol. 31, no 2, p. 396-408.

[19] Mangayarkarasi, v., durairaj, elantamilan, ramanathan, vijaya, et al. Enhancing cancer screening and early diagnosis in india: overcoming challenges and leveraging emerging technologies. Cureus, 2025, vol. 17, no 2.

[20] Meddaoui, mohamed amine. Semantic web for sharing medical resources. In : international conference on business intelligence. Cham : springer international publishing, 2021. P. 388-398.

[21] Su, weihang, tang, yichen, ai, qingyao, et al. Parametric retrieval augmented generation. In : proceedings of the 48th international acm sigir conference on research and development in information retrieval. 2025. P. 1240-1250.

[22] Dossouvi, komla mawunyo, sellera, fábio parra, ibadin, ephraim ehidiamen, et al. Recommendations and adjuvant therapies to mitigate clarithromycin-resistant helicobacter pylori-associated infections in africa. Infection and drug resistance, 2025, p. 3169-3174.

[23] Li, shuiquan, chen, xi, shi, hui, et al. Tailoring traditional chinese medicine in cancer therapy. Molecular cancer, 2025, vol. 24, no 1, p. 27.

[24] Dhandapani, gokila, n, porchelvi, u, arun kumar, et al. Design and development of graphene-based c-shaped silicon-mgf2-silicon materials surface plasmon resonance biosensor for cervical cancer detection. Plasmonics, 2025, p. 1-11.

[25] Cicatiello, annunziata gaetana, musone, michele, imperatore, silvestro, et al. Circulating mirnas in genitourinary cancer: pioneering advances in early detection and diagnosis. The journal of liquid biopsy, 2025, vol. 8, p. 100296.

[26] Zafar, aasma, khatoon, summaiya, khan, muhammad jawad, et al. Advancements and limitations in traditional anti-cancer therapies: a comprehensive review of surgery, chemotherapy, radiation therapy, and hormonal therapy. Discover oncology, 2025, vol. 16, no 1, p. 607.

[27] Al amin, md, liza, irin akter, hossain, shah foysal, et al. Enhancing patient outcomes with ai: early detection of esophageal cancer in the usa. Journal of medical and health studies, 2025, vol. 6, no 1, p. 08-27.

[28] Noreen, sana, ishaq, izwa, saleem, muhammad hamzah, et al. Electrochemical biosensing in oncology: a review advancements and prospects for cancer diagnosis. Cancer biology & therapy, 2025, vol. 26, no 1, p. 2475581.

[29] Yu, hai-jia, liu, jian-hua, liu, wei, et al. Metal-based nanomedicines for cancer theranostics. Military medical research, 2025, vol. 12, no 1, p. 41.

[30] Dai, charles s., mishra, avanish, edd, jon, et al. Circulating tumor cells: blood-based detection, molecular biology, and clinical applications. Cancer cell, 2025.

[31] Meddaoui, mohamed amine, erritali, mohammed, et al. A comparative study of embedded learning models iot-based for real time mask detection. International journal of advanced science and computer applications, 2024, vol. 3, no 2.

[32] El-saleh, ayman a., sheikh, abdul manan, albreem, mahmoud am, et al. The internet of medical things (iomt): opportunities and challenges. Wireless networks, 2025, vol. 31, no 1, p. 327-344.

[33] Mansouri, wahida, alohali, manal abdullah, alqahtani, hamed, et al. Deep convolutional neural network-based enhanced crowd density monitoring for intelligent urban planning on smart cities. Scientific reports, 2025, vol. 15, no 1, p. 5759.

[34] Jin, yuebo et huang, yadong. Long-term vital sign tracking study of depression patients based on wearable devices. International journal of crowd science, 2025, vol. 9, no 1, p. 56-63.

[35] Unanah, onyekachukwu victor et aidoo, esi mansa. The potential of ai technologies to address and reduce disparities within the healthcare system by enabling more personalized and efficient patient engagement and care management. World journal of advanced research and reviews, 2025, vol. 25, no 2, p. 2643-64.

[36] Dong, chaoqun, ji, yan, fu, zhongmin, et al. Precision management in chronic disease: an ai empowered perspective on medicine-engineering crossover. Iscience, 2025, vol. 28, no 3.

[37] Koontalay, apinya, botti, mari, et hutchinson, anastasia. Development of a user - centred chronic care model for patients with heart failure in a limited - resource setting: a codesign study. Health expectations, 2025, vol. 28, no 1, p. E70142.

[38] Shen, shiying, qi, wenhao, zeng, jianwen, et al. Passive sensing for mental health monitoring using machine learning with wearables and smartphones: scoping review. Journal of medical internet research, 2025, vol. 27, p. E77066.

[39] Bladon, siân, eisner, emily, bucci, sandra, et al. A systematic review of passive data for remote monitoring in psychosis and schizophrenia. Npj digital medicine, 2025, vol. 8, no 1, p. 62.

[40] Fritz, roschelle, cook, diane, et al. Detecting older adults' behavior changes during adverse external events using ambient sensing: longitudinal observational study. Jmir nursing, 2025, vol. 8, no 1, p. E69052.

[41] Wani, m. Arif, ali, sarwat, sofi, mukhtar ahmad, et al. Advances in deep learning, volume 2. Springer nature, 2025.

[42] Ali, muhammad, benfante, viviana, basirinia, ghazal, et al. Applications of artificial intelligence, deep learning, and machine learning to support the analysis of microscopic images of cells and tissues. Journal of imaging, 2025, vol. 11, no 2, p. 59.

[43] Meddaoui, mohamed amine, amzil, moulay, karkaba, imane, et al. Deep learning optimization conception: less data, less time, more performance. International journal of advanced computer science & applications, 2025, vol. 16, no 7.

[44] Karthik, a., sahoo, santosh kumar, kumar, ajay, et al. Unified approach for accurate brain tumor multi-classification and segmentation through fusion of advanced methodologies. Biomedical signal processing and control, 2025, vol. 100, p. 106872.

[45] V. Z. Kuklin, l. M. Chervyakov, a. N. Ruslantsev, and t. G. Aslanov, "technique for segmentation and selection of biomedical images", j. Hum. Earth future, vol. 5, no. 4, pp. 688–703, dec. 2024

[46] Kim, dong won, seo, jihoon, kwon, sujin, et al. Predicting in - hospital fall risk using machine learning with real - time location system and electronic medical records. Journal of cachexia, sarcopenia and muscle, 2025, vol. 16, no 1, p. E13713.

[47] Abbasi, fereshteh, naderan, marjan, et alavi, seyed enayatallah. Dimensionality reduction with deep learning classification for botnet detection in the internet of things. Expert systems with applications, 2025, vol. 267, p. 126149.

[48] S. Armoogum, k. Motean, d. A. Dewi, t. B. Kurniawan, and j. Kijsomporn, "breast cancer prediction using transfer learning-based classification model", emerg sci j, vol. 8, no. 6, pp. 2373–2384, dec. 2024.

[49] Gao, ding, zhi, yuan, rong, xing, et al. Mismatch analysis of rooftop photovoltaics supply and farmhouse load: data dimensionality reduction and explicable load pattern mining via hybrid deep learning. Applied energy, 2025, vol. 377, p. 124520.

[50] Vinoth, n. A. S., kalaivani, j., arieth, r. Madonna, et al. An enhanced fusion of transfer learning models with optimization based clinical diagnosis of lung and colon cancer using biomedical imaging. Scientific reports, 2025, vol. 15, no 1, p. 24247.

[51] S. M. N. Nobel, o. F. Sifat, m. R. Islam, m. S. Sayeed, and m. Amiruzzaman, "enhancing gi cancer radiation therapy: advanced organ segmentation with reseca-u-net model", emerg sci j, vol. 8, no. 3, pp. 999–1015, jun. 2024.

[52] Meddaoui, mohamed amine, erritali, mohammed, madani, youness, et al. Mask detection using iot-a comparative study of various learning models. In : international conference on smart homes and health telematics. Cham : springer international publishing, 2022. P. 272-283.

[53] Eita, mohammad a. Et rizk, hamada. A trustworthy and explainable deep learning framework for skin lesion detection in smart dermatology. Engineering applications of artificial intelligence, 2025, vol. 159, p. 111594.

[54] Asif, sohaib, wenhui, yi, ur-rehman, saif-, et al. Advancements and prospects of machine learning in medical diagnostics: unveiling the future of diagnostic precision. Archives of computational methods in engineering, 2025, vol. 32, no 2, p. 853-883.

[55] Gurcan, fatih. Enhancing breast cancer prediction through stacking ensemble and deep learning integration. Peerj computer science, 2025, vol. 11, p. E2461.

[56] Fabbrocini, gabriella, triassi, maria, mauriello, maria chiara, et al. Epidemiology of skin cancer: role of some environmental factors. Cancers, 2010, vol. 2, no 4, p. 1980-1989.

[57] Leadbeater, ellouise et watrobska, cecylia. What could evolve in the evolution of memory?. Philosophical transactions b, 2025, vol. 380, no 1929, p. 20240109.

[58] Locatelli, pierluigi, et al. Investigating secure and distributed control in iot: improving ble security and strengthening lorawan with blockchain. 2025.

[59] Wang, jing, quasim, mohammad tabrez, et yi, bo. Privacy-preserving heterogeneous multi-modal sensor data fusion via federated learning for smart healthcare. Information fusion, 2025, vol. 120, p. 103084.

[60] Reicher, lee, shilo, smadar, godneva, anastasia, et al. Deep phenotyping of health–disease continuum in the human phenotype project. Nature medicine, 2025, p. 1-13.

[61] Desai, priyanka, karthik, p., loganathan, d., et al. Different data cleaning techniques and normalization techniques with focus on current normalization techniques: a study. In : advances in electrical and computer technologies. Crc press, 2025. P. 332-349.

[62] Tobieha, atefeh, behzadfar, neda, yousefi, mohammadreza, et al. Analysing and investigating the effect of heroin use disorder for the changes of distinguishing features in brain signal processing. Iet science, measurement & technology, 2025, vol. 19, no 1, p. E70023.

[63] Asgari mehrabadi, milad, khatibi, elahe, jimah, tamara, et al. Perfect: personalized exercise recommendation framework and architecture. Acm transactions on computing for healthcare, 2025, vol. 5, no 4, p. 1-25.

Comparative Analysis of Machine Learning and Deep Learning Models for Handwritten Digit Recognition

Soukaina Chekki, Boutaina Hdioud, Rachid Oulad Haj Thami, Sanaa El Fkihi
Advanced Digital Enterprise Modeling and Information Retrieval (ADMIR) Laboratory-Rabat IT Center, Information Retrieval and Data Analytics Team (IRDA), ENSIAS, Mohammed V University in Rabat, Rabat, Morocco

***Abstract*—Handwritten digit recognition (HDR) forms a key component of computer vision systems, especially in optical character recognition (OCR). This study presents a comparative analysis of Machine Learning (ML) algorithms and Deep Learning (DL) models for HDR tasks. A contour-based segmentation technique was applied in preprocessing to enhance feature extraction by detecting digit boundaries and reducing noise. ML models, including K-Nearest Neighbors (KNN) and Support Vector Machine (SVM), and DL architectures, such as Artificial Neural Networks (ANNs), Convolutional Neural Networks (CNNs), and Recurrent Neural Networks (RNNs), were evaluated on the Modified National Institute of Standards and Technology (MNIST) and the National Institute of Standards and Technology (NIST) datasets. The results demonstrate that DL models significantly outperform ML algorithms in terms of accuracy and robustness, while the KNN model achieved acceptable results. The results underline the importance of contour-based preprocessing in boosting deep learning techniques for HDR.**

Keywords*—Handwritten digit recognition (HDR); Optical Character Recognition (OCR); Machine Learning (ML); Deep Learning (DL); segmentation

I. Introduction

Handwritten text recognition has long been a challenging and active research field, particularly due to the reliance on handcrafted features and prior knowledge in traditional optical character recognition (OCR) systems. Handwritten digit recognition (HDR) has attracted continuous attention over the decades, with advances in computing, artificial intelligence, and imaging since the 1960s driving progress [1], [2], [3], [4]. Although recent developments in deep learning have significantly improved recognition accuracy, the growing demand for large datasets and computational resources continues to motivate research toward more efficient solutions. Robust methods exist for several languages such as English, Chinese, and Japanese [5], [6], [7], [8], [9]; however, recognition of highly variable handwritten digits remains a significant challenge.

This study focuses on offline HDR, where variability in handwriting, overlapping digits, and noise make segmentation a critical step. In particular, separating connected digits remains a difficult problem that directly impacts recognition performance. To address these challenges, we propose a contour-based segmentation approach combined with machine learning and deep learning models. Five models were evaluated: Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Artificial Neural Network (ANN), Recurrent Neural Network (RNN), and Convolutional Neural Network (CNN), all widely used in supervised learning and image-based pattern recognition.

Our system was trained on the MNIST dataset and tested on the NIST dataset, which contains numerous complex examples with challenging segmentation cases, making it a particularly suitable benchmark for evaluation. Morphological and geometric operations were integrated to enhance robustness against noisy handwritten images. The proposed approach achieved high recognition accuracy, demonstrating the effectiveness of combining contour-based segmentation with ML and DL techniques.

This work contributes to HDR research by providing a comparative evaluation of machine learning and deep learning approaches while highlighting the critical role of segmentation in offline recognition systems.

The study is organized as follows: Section II reviews related work. Section III presents the proposed methodology. Section IV describes the approaches used for recognizing handwritten digits, including details of all models employed. Section V presents experimental results, and Section VI concludes the study and outlines future work.

II. Related Work

HDR remains a central topic in machine learning due to its practical relevance and the inherent variability of human handwriting. Recent approaches increasingly rely on data-driven methods that integrate traditional algorithms with deep neural networks to enhance recognition performance. Achieving accurate character recognition from image inputs requires effective techniques, particularly in segmentation-based methods [10], [11].

HDR has been extensively investigated using both traditional ML and modern DL techniques. Early and recent studies have explored handcrafted feature extraction methods for HDR. Techniques such as Histogram of Oriented Gradients (HOG), Local Binary Patterns (LBP), and Gabor filters were widely used to extract discriminative image features. Classifiers including K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Random Forest (RF), and Gradient-Boosted Decision Trees (GBDT) were then applied. For instance, combining HOG features with SVM achieved an accuracy of 93.32% on the NumtaDB dataset [12], highlighting the moderate yet consistent performance of feature-based ML approaches.

Transfer learning using pre-trained CNNs has also been applied, particularly for non-Latin scripts such as Bengali digits.

Networks including ResNet-50, Inception-v3, and EfficientNetB0 were fine-tuned on the NumtaDB dataset containing 17,000 instances across ten classes. These models achieved 97% accuracy, demonstrating the benefits of leveraging pre-trained representations to manage handwriting variability and limited data scenarios [13].

However, more recent studies have reported slightly lower accuracy, reflecting ongoing challenges in HDR. For example, one study reported 97% accuracy on the NumtaDB dataset for Bangla handwritten digits using a pre-trained CNN model [14], [15], while another study achieved 96.9% accuracy with the YOLOv8x model for digit detection [16].

Recent work has further investigated hybrid and ensemble approaches. Hybrid models combining CNN-based feature extraction with SVM classification, as well as ensembles integrating CNNs with traditional ML classifiers, have achieved higher accuracy, up to 99.30%, on benchmark datasets such as MNIST [17], [18]. Additionally, optimization strategies, including mini-batch and Hessian-free training techniques, have been employed to enhance computational efficiency and generalization [19].

In comparison, limited work has been conducted on recognizing handwritten Farsi digits because of their cursive structure and variation in shape, which differ considerably from Latin-based scripts [20], [21]. A key distinction between the Farsi and Arabic languages is the right-to-left writing orientation, which introduces further complexity to the segmentation and recognition tasks [22], [23], [24].

Despite these advances, limitations remain, particularly regarding segmentation robustness and the handling of complex scripts. These challenges motivate the proposed contour-based segmentation approach, which integrates both machine learning and deep learning models. By addressing overlapping and noisy digits, the system aims to improve recognition accuracy on challenging datasets, including MNIST for training and the more complex NIST dataset for testing. Consequently, this work contributes to overcoming current HDR limitations and strengthening offline recognition capabilities.

III. Methodology

HDR systems rely on preprocessing, feature extraction, and classification. This study assesses SVM, KNN, ANN, RNN, and CNN within a contour-based pipeline to identify the best model.

A. General Architecture

Handwritten digit recognition systems are generally based on the following steps, as shown in Fig. 1.

B. Image Acquisition

The acquisition phase captures handwritten text using devices like scanners or digital cameras, converting it into digital images (e.g., JPEG, PNG) with minimal information loss. These images are then preprocessed to improve quality, correct distortions, and remove noise, preparing them for reliable analysis in tasks such as optical character recognition (OCR).

C. Image Preprocessing

Preprocessing prepares images for recognition by applying binarization, noise reduction, normalization, and slant/skew correction [25]. It may also extract handwritten fields and remove backgrounds to isolate the target content.

1) Binarization (Thresholding): Applies either global or adaptive thresholding based on pixel intensity.

2) Noise reduction and normalization: Uses filtering and morphological operations. Normalization corrects slant, skew, and size to reduce variability and improve accuracy [26], [27].

3) Mathematical morphology: Employs operations like erosion, dilation, opening, closing, and morphological gradients for noise removal and contour enhancement.

D. Segmentation

This phase aims to separate graphemes, words, or numerical strings. Segmenting handwritten characters is one of the most challenging tasks in text recognition. A typical application is the recognition of numerical amounts in bank checks, which is particularly difficult because of the presence of noise, overlapping digits, and even the fragmentation of digits into multiple disconnected components, making them difficult to identify [28], [29].

E. Feature Extraction

Feature extraction is vital in character recognition, as classifiers cannot compensate for poor features. It captures relevant information to reduce within-class variability and enhance class separability, yielding numerical feature vectors [30]. Extracted features are generally classified as structural, statistical, global (e.g., transformations), or topological/metric features [31], [32].

F. Classification

After segmentation and feature extraction, classification assigns inputs to classes based on feature vectors, typically through supervised learning involving training and recognition stages. Outcomes include successful classification, confusion between similar classes, or rejection when no adequate match exists. Classifiers vary in performance and robustness, with main types including shape matching, structural methods, neural networks, and statistical classifiers [33].

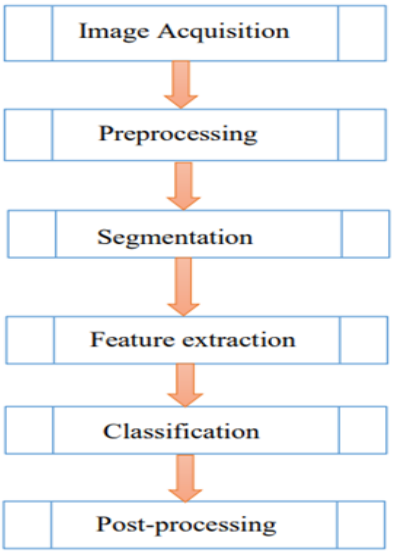


Fig. 1. General diagram of a handwritten digit recognition system.

G. Post-Processing

Post-processing refining recognition results through linguistic, contextual, and structural analysis. It includes:

1) Lexical verification: Checks recognized words against a dictionary, often with edit distance metrics.

2) Orthographic correction: uses n-gram models to correct likely letter sequence errors, such as "tble" to "table".

3) Syntactic validation: Ensures grammatical correctness and suggests corrections [34].

4) Semantic adaptation: Validates word meaning based on context.

5) Pragmatic adaptation: Detects anomalies by comparing recognized words to expected formats.

Post-processing is essential for handling recognition failures, especially those caused by upstream errors or limited training data [35].

IV. Approaches Used in Recognizing Handwritten Digits

Various classifiers are used in OCR development, including parametric, non-parametric statistical, and hybrid methods. In this study, after segmenting the relevant images, we applied ML and DL models such as SVM, KNN, ANN, RNN, and CNN on segmented images from the NIST dataset to recognize handwritten digits. Each model was selected for its strengths in handling different data aspects. Performance was evaluated using accuracy, precision, recall, F1-score, confusion matrix, and ROC/AUC, allowing clear comparison to identify the best model.

A. Machine Learning

ML allows computers to learn from data without explicit programming [36]. In handwritten digit recognition, ML typically requires feature extraction before classification. Traditional algorithms such as SVM, decision trees, and Bayesian networks are flat models that depend heavily on feature quality.

1) Support Vector Machine (SVM): SVMs are effective classifiers in HDR, achieving up to 93,32 % accuracy using projection histograms and HOG [37]. As illustrated in Fig. 2, HDR has posed a significant challenge since the 1970s. Its importance lies in applications, such as writer identification, automated data extraction from scanned documents, and various other domains that require reliable character interpretation.

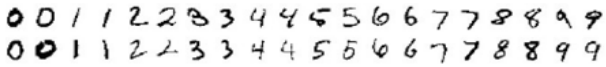


Fig. 2. Digits to recognize [38].

This study aims to accurately identify digits amid symbols and noise, despite overlapping class boundaries caused by similar digit shapes.

Formally, the objective function to be minimized in the context of SVM is computed, as shown in Eq. (1):

$$f(\varepsilon, w) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\varepsilon_i \quad (1)$$

where,

- ε_i represents the slack variables for misclassified samples (i.e., points located on the incorrect side of the separating hyperplane).
- $\boldsymbol{C}$ is a regularization parameter that balances the trade-off between maximizing the margin and minimizing classification error.
- $\boldsymbol{w}$ denotes the weight vector that defines the hyperplane orientation.

In such cases, margin-based methods that maximize the distance between the support vectors and the hyperplane are preferred. As illustrated in Fig. 3, this involves optimizing two margin boundaries (e.g., defined by vectors w_1 and w_2) to ensure robust separation, even in the presence of overlapping classes.

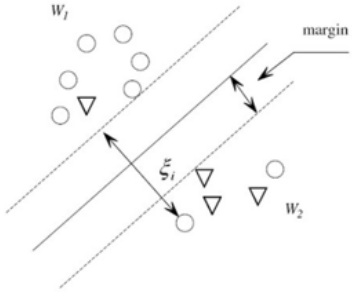


Fig. 3. Feature space accompanied by the margin of error [38].

```
accuracy: 0.9348412698412698

[[1156    0    5    1    0    2   12    6    5    1]
 [   0 1389    4    3    4    0    2    6    5    0]
 [   2    4 1170   19    5    0    9   35   15    1]
 [   1    4   19 1237    0   20    2   32   13    3]
 [   0    4   25    0 1133    0    9   17    3   27]
 [   1    9    5   42    3 1047   17   11    8    6]
 [  13    2    5    0    2   15 1156   24    2    0]
 [   3   10    9    3   15    0    0 1256    1   33]
 [   5   19   14   25    7   21    8   15 1095    8]
 [   4    4    6   30   23    2    1   61    4 1140]]
```

Fig. 4. Confusion matrix of our SVM model.

Fig. 4 shows the SVM confusion matrix on the MNIST dataset, with 93.48% accuracy. High diagonal values indicate strong overall performance, though some misclassifications occur, notably for digit '4'. Performance improvements can be achieved via data augmentation, hyperparameter tuning, feature engineering, or hybrid classifiers. The following sections compare these results with deep learning models on the same dataset.

2) K-Nearest Neighbors (KNN): Among the various classification methods, the KNN is notable for its simplicity and effectiveness. It classifies new observations by identifying the closest instances in the training dataset and assigning the most common class to them [39].

KNN retains all available training examples and uses a distance-based similarity metric to perform classification. It is widely recognized as a *non-parametric lazy learning* algorithm [40] (see Algorithm 1).

Algorithm 1 KNN Algorithm (K-Nearest Neighbors)

Input: A labeled training set $L = \{(x', c)\}$, and a new instance x

Output: Predicted class label for x

1. **For** each example $(x', c) \epsilon L$
   Calculate the distance $D(x, x')$
   End for
2. Select the k nearest neighbors of x, denoted as $kNN(x)$, based on the smallest distances
3. **For** each $x' \epsilon kNN(x)$
   count the occurrences of each class
   End for
4. Assign x the class with the highest frequency among neighbors

This approach measures similarity between test samples and training data, classifying based on a threshold from the k-nearest neighbors. KNN's performance depends on the number of neighbors, similarity metric, threshold, and data representativeness [41]. Trained and tested on MNIST, the Scikit-learn implementation used six neighbors with uniform weights.

B. Deep Learning

1) Artificial Neural Networks (ANN): Fig. 5 illustrates the architecture of the proposed ANN, which consists of an input layer, hidden layer, and output layer [42]. The input layer is connected to the hidden layer through weighted links, denoted by W_{ij}, where i refers to the input neuron and j to the hidden neuron. Similarly, the connections between the hidden and output layers are represented by the weights W_{jk}, where k indicates the output neuron.

A bias term of +1 was integrated into the network to enhance the flexibility of the parameter adjustment. The structure follows a multi-layer perceptron (MLP) configuration.

In this model, the sigmoid activation function was applied to both hidden and output layers. As shown in Eq. (2), the sigmoid maps input values to the [0, 1] range, allowing smooth, bounded output transitions and supporting effective training.

$$g(x) = \frac{1}{1+e^{-x}} \quad (2)$$

In this context, $g(x)$ denotes the sigmoid activation function, and x represents the net input, that is the weighted sum of the inputs to a neuron.

2) Réseaux De Neurones Récurrents (RNN) : Unlike MLPs, RNNs (Fig. 6) have cycles in their computational graphs, enabling memory retention [43]. Unrolled over time (Fig. 7), they form chains of MLPs with recurrent links, capturing temporal dependencies, which makes them well-suited for HDR tasks.

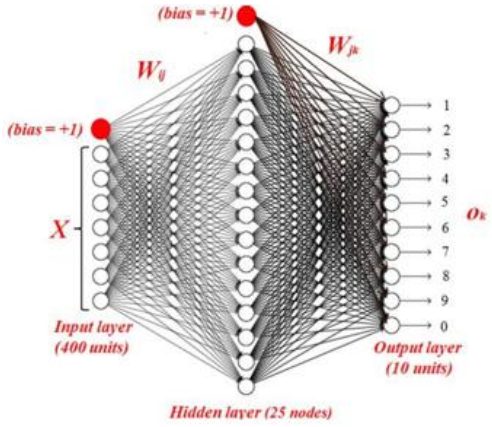


Fig. 5. The suggested architecture for the neural network.

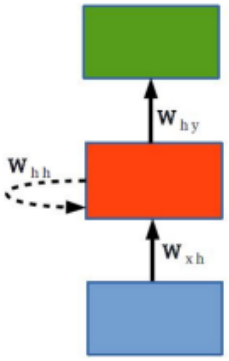


Fig. 6. Compact representation of RNN. All arrows depict complete connections. The dashed arrow represents connections with a temporal shift (t − 1).

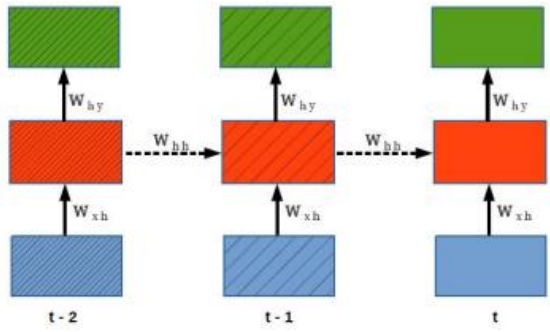


Fig. 7. Unfolded representation of RNN [43].

According to this model, an RNN takes a sequence of events $x = (x1, x2, \dots, xT)$ as the input and defines the sequence of hidden states $h = (h1, h2, \dots, hT)$ to produce the sequence of output vectors $y = (y1, y2, \dots, yT)$ by iterating from t=1 to T :

$$h_t = H(W_{x_h} x_t + W_{h_h} h_{t-1} + b_h) \quad (3)$$

$$y_t = W_{h_y} h_t + b_y \quad (4)$$

where, T is the total number of input vectors, $W_{\alpha\beta}$ is the weight matrix between layers α and β, and $b\beta$ is the bias vector in layer β. The activation function H used in RNNs is generally a hyperbolic tangent (tanh).

Traditional backpropagation, designed for non-sequential networks, cannot capture temporal dependencies, as shown in Eq. (3) and Eq. (4). RNNs address this limitation by unrolling over time, enabling Backpropagation Through Time (BPTT) to train the network effectively [44].

The distinctive feature of this representation, as opposed to a 'classic' non-recurrent neural network, is the presence of multiple shared parameters. For instance, a common weight matrix W_{h_h} conveys information through diagonal connections. Additionally, the weight matrices, W_{x_h} and W_{h_y} depicted by the vertical connections, are shared across time steps.

3) Convolutional Neural Network (CNN): CNNs have gained popularity in recent years due to their effectiveness in image recognition. A typical CNN includes convolutional, pooling, and fully connected layers, often with ReLU activation. Training involves a feed-forward phase to extract features and a back-propagation phase for weight updates [45].

In the feed-forward phase, an input image passes through the network, undergoing convolutional operations where neurons apply learned filters to extract features. This process transforms the input through successive layers until the final output is produced [46], [47].

As shown in Fig. 8, a CNN consists of an input layer, an output layer, and multiple hidden layers, including convolutional, pooling (e.g., max and average pooling), normalization, and fully connected layers. Filters extract spatial features, while non-linear activations add complexity. As data flows through the network, spatial dimensions shrink and feature channels grow, leading to the final prediction.

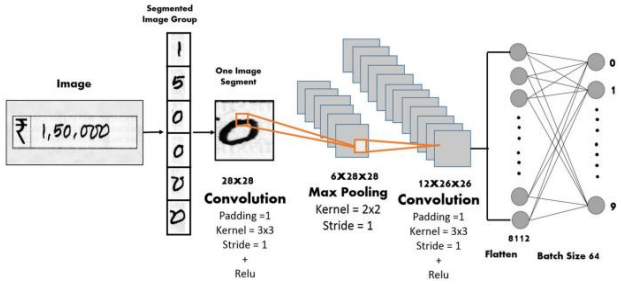


Fig. 8. CNN architecture for handwritten amount digit recognition [47].

V. Implementation and Results

The primary objective of this study was to implement and evaluate various models and architectures for HDR. Initially, contour-based segmentation was applied to extract the digit regions from the input images. Subsequently, a range of machine learning and deep learning models, namely, SVM, KNN, ANN, CNN, and RNN, were employed to assess their performance and determine the most effective approach for this recognition task.

In this study, the models were trained on the MNIST dataset and tested on the NIST dataset, since NIST contains handwritten digits that are more challenging to classify. This setup was designed to evaluate the cross-dataset generalization capability of the proposed approach. The differences in style and noise between the two datasets represent a real challenge, and our methodology highlights the robustness of the proposed solution under such conditions. Furthermore, we discuss the potential of applying domain adaptation techniques to further enhance performance on the NIST dataset.

A. Dataset Description

1) MNIST dataset: Fig. 9 presents the MNIST dataset, a benchmark collection of 70,000 grayscale images of handwritten digits compiled by NIST. Each 28×28 image represents one of ten digit classes (0 to 9). The dataset is split into 60,000 training and 10,000 testing samples, each labeled accordingly.

MNIST was selected for this study due to its standardized format, broad use in literature, and relevance for offline HDR. Its consistency, ease of use, and compatibility with various machine learning models make it ideal for performance evaluation.

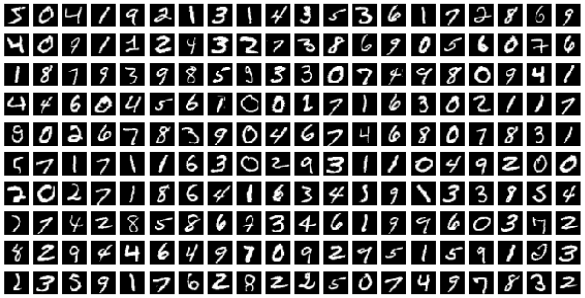


Fig. 9. Handwritten digits from the MNIST database.

2) NIST dataset: The specialized database illustrated in Fig. 10 encompasses the complete training dataset used by NIST for the recognition of handwritten text and characters in printed documents. It contains handwritten samples contributed by 3,600 writers and includes 810,000 isolated character images extracted from these forms. The dataset also provides ground-truth classifications for each image, reference templates to support future data-collection efforts, and a suite of software tools for image processing and annotation.

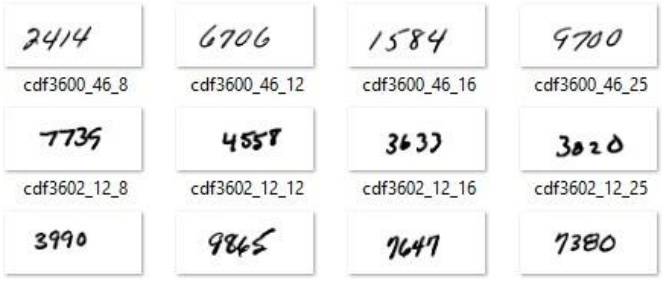


Fig. 10. Handwritten digits from the NIST database.

B. Contour Segmentation

Contour-based segmentation proposed in this study is more effective than classical methods. Our approach implements multiple specific steps for digit segmentation, which allows better isolation of each character, reduces noise impact, and improves the quality of segmented regions. Unlike standard methods that use simple binarization or contour detection, our

multi-step process preserves digit integrity and enhances performance, particularly in cross-dataset generalization trained on MNIST and tested on NIST.

This section presents handwritten digit recognition using ML and DL models, including KNN, SVM, ANN, RNN, and CNN. MNIST was used for training, NIST for testing, and OpenCV for preprocessing the digit images.

Steps in the Handwritten Digit Recognition Pipeline Using Contour Segmentation:

1) Image acquisition: Handwritten digit images were collected from the MNIST dataset.

2) Dataset splitting: The data were separated into a training set (MNIST) and a testing set (NIST).

3) Preprocessing: Contour-based segmentation techniques were applied to clean and prepare the images.

4) Normalization: Pixel intensity values were scaled to a range between 0 and 1.

5) Batching: The training dataset was divided into mini-batches to enhance training efficiency.

6) Model training: ML and DL models were trained using labeled data.

7) Classification: The trained models were used to classify digits in the testing dataset.

8) Performance evaluation: Recognition accuracy and processing time were evaluated for each model variant.

C. Implementations

1) Image preprocessing: Fig. 11 illustrates a sample image from the NIST database, which may exhibit noise, spots, or other unwanted artifacts. To mitigate these imperfections, a Gaussian filter was applied to reduce the noise, followed by a derivative filter for edge detection. A sharpening technique based on a blur mask was employed to enhance image clarity, as shown in Fig. 12.



Fig. 11. Initial image.

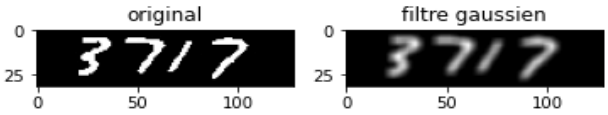


Fig. 12. Gaussian smoothing on the image.

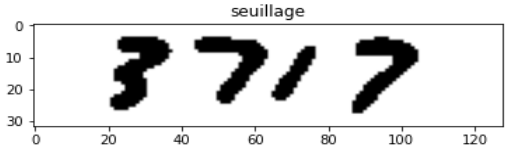


Fig. 13. Thresholding "seuillage" application.

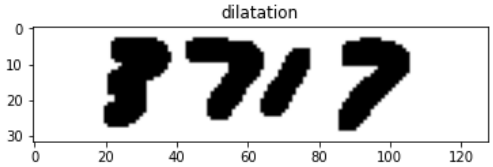


Fig. 14. Applying dilation to the image.

Thresholding was applied, as illustrated in Fig. 13, using Otsu's method to determine the optimal global threshold based on the image histogram. This binary thresholding approach enhances the contrast between foreground and background. As shown in Fig. 14, dilation was selectively applied to images containing small handwritings to capture their contours more effectively. In addition, the Canny edge detector, known for its effectiveness in edge detection, was employed in the preprocessing phase, as illustrated in Fig. 15.

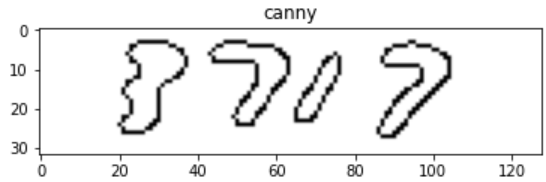


Fig. 15. Application of Canny Edge Detection.

2) Contour segmentation application: In the segmentation phase, we applied a contour detection algorithm to isolate individual digits within the image. A contour is defined as a closed curve that connects all the continuous points along the boundary of an object, representing its shape, as illustrated in Fig. 16. To accomplish this, one function was used to locate the contours, and the other was used to draw them.

Subsequently, as shown in Fig. 17, we combined contour-based segmentation with vertical segmentation. The latter divides the image into vertical slices to separate connected digits, thereby facilitating more accurate digit recognition.

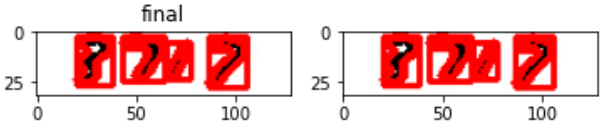


Fig. 16. Contour segmentation on the NIST dataset image digits.



Fig. 17. Vertical segmentation application.

D. Model hyperparameters

The CNN model includes a convolutional layer with 32 filters and ReLU activation, followed by a flattening layer, and then two dense layers: the first with 100 units and ReLU activation, and the second with 10 units and softmax activation. We used the SGD optimizer (learning rate of 0.01, momentum of 0.9) and the 'categorical_crossentropy' loss function, as shown in Table III.

TABLE I. ANN HYPERPARAMETERS

Number of epochs	100
Batch size	1000
Activation function	Softmax & Sigmoid
Optimizer	Adam
Loss	sparse_categorical_crossentropy

TABLE II. RNN HYPERPARAMETERS

Number of epochs	50
Batch size	128
Units	256
Dropout	0.2
Activation function	Softmax
Optimizer	SGD
Loss	categorical_crossentropy

TABLE III. CNN HYPERPARAMETERS

Number of epochs	100
Batch size	1000
Activation function	Relu & Softmax
Optimizer	Adam
Loss	categorical_crossentropy

As illustrated in Table I, the ANN model was trained for 100 epochs with a batch size of 1000 using Softmax and Sigmoid activation functions and the Adam optimizer, with 'sparse_categorical_crossentropy' as the loss function.

To prevent overfitting, the RNN model, which includes 256 units and a dropout rate of 0.2 to prevent overfitting, uses Softmax activation and is optimized with SGD, as shown in Table II. The loss function used is also 'categorical_crossentropy', which is optimized for multi-class classification tasks.

The hyperparameters for all models were selected based on experimental validation to ensure stable convergence, prevent overfitting, and achieve optimal performance on both MNIST and NIST datasets. Ablation studies were performed to evaluate the impact of varying key parameters (e.g., number of units, activation functions, learning rate, dropout), confirming that the chosen configuration consistently provides robust and reproducible results.

E. Analysis of Results

During testing, this study proposes a comprehensive method to evaluate all digit cases from the NIST dataset across multiple models. A structured framework was developed to facilitate testing on different models by grouping each test case within a structure that consolidates the results of all applied models. Unlike previous studies, which performed tests on the entire dataset without a method allowing an easy linkage of each model to its corresponding test, this approach ensures clear, systematic, and robust performance comparisons, as illustrated below. This methodology confirms the relevance of the observed improvements.

We performed handwritten digit recognition using five different models (KNN, SVM, CNN, RNN, and ANN) to identify the most accurate model. As shown in Fig. 18, the deep learning models (ANN, RNN, and CNN) outperformed the others in both test cases. Similarly, Fig. 19 shows that the RNN and CNN achieved the best results across all the digits in the test case. Furthermore, Fig. 20 illustrates that the RNN consistently outperformed the CNN and other models across all three test scenarios.

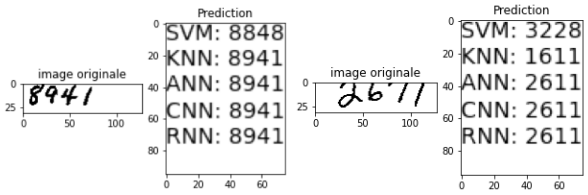


Fig. 18. In both test cases, Deep Learning models {ANN, RNN, CNN} showed superior performance.

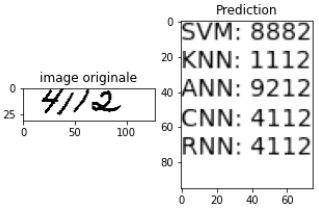


Fig. 19. RNN and CNN models achieved the best results for all digits in this test.

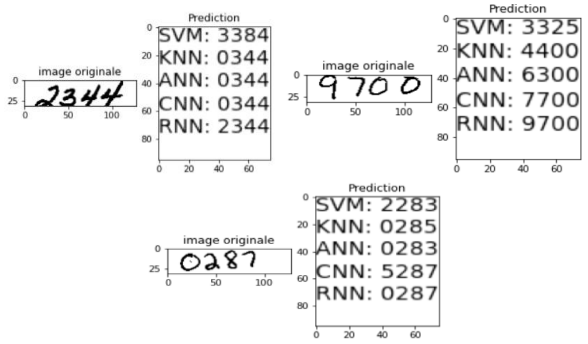


Fig. 20. RNN models outperformed CNN and other models across all digits in these three tests.

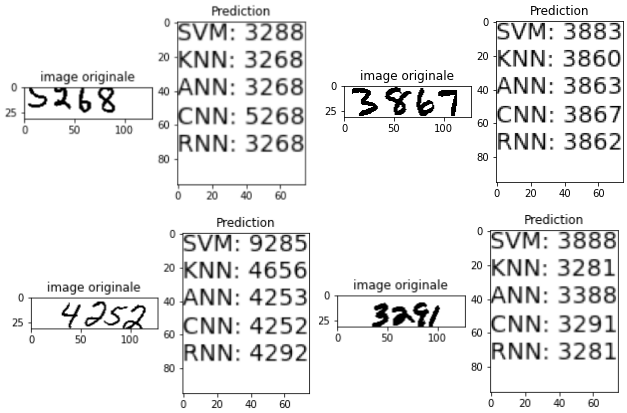


Fig. 21. CNN models performed better than other models for all digits in these examples.

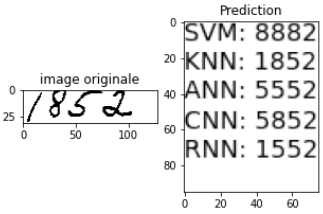


Fig. 22. In this test example, KNN models outperform others and correctly identify at least 3 out of 4 digits in other tests.

In the test examples shown in Fig. 21, the CNN models outperformed the other models for all the digits. As shown in Fig. 22, the KNN models also demonstrated superior performance for all the digits. In addition, in other test examples, the KNN models provided acceptable results, correctly identifying at least three of the four handwritten digits.

TABLE IV. RESULTS TABLE DISPLAYING THE ACCURACY OF EACH MODEL

Models	Accuracy (%)
SVM	93.48
KNN	92.9
ANN	99.56
RNN	98.74
CNN	98.53

Table IV lists the accuracy results for each model on the MNIST image dataset. In our handwritten digit recognition project using the MNIST dataset, we evaluated several ML and DL models: SVM, KNN, ANN, CNN, and RNN. The results showed that the DL models outperformed the traditional machine learning models. In particular, ANN achieved the highest accuracy of 99.56%, followed by RNN at 98.74% and CNN at 98.53%. SVM and KNN achieved accuracies of 93.48% and 92.9%, respectively. Prediction analysis for the digit sequence '1852' highlighted KNN's strength in certain specific cases, whereas other models had variable prediction accuracies. Overall, the DL models proved to be more effective for this task.

TABLE V. COMPARISON OF RESULTS WITH RECENT WORKS

Models	Our Result (%)	Previous Results (%)	References of Previous Works
SVM	93.48	92.3- 93	[48]
KNN	92.9	91.8- 92.3	[49]
ANN	99.56	97 - 97.4	[50]
RNN	98.74	97 - 98	[51]
CNN	98.53	98 – 98.12	[52]

Our model outperformed previous approaches across all tested algorithms, as shown in Table V, with significant improvements in the ANN (99.56%) and RNN (98.74%). These enhancements suggest effective optimization of the architecture and hyperparameters. Slight increases were also observed for SVM, KNN, and CNN, confirming the robustness of our approach compared to previous works.

VI. CONCLUSION AND FUTURE WORK

This study developed an offline handwritten digit recognition system and compared various ML and DL models to identify the most effective. After preprocessing, a contour-based segmentation method was implemented and evaluated across multiple classifiers. The study assessed features and models to determine the best HDR approach.

Segmentation is challenging. Our system uses morphological and geometric operations with contour detection to isolate digits and reduce noise, producing segmented images of single digits for training.

We performed experiments on the NIST dataset using contour-based segmentation and a range of algorithms: KNN, SVM, ANN, CNN, and RNN. The results showed that CNN and RNN delivered the best recognition performance, with high accuracy and consistency. Among the ML methods, KNN achieved acceptable results despite the difficulty in recognizing unclear digits in the NIST dataset.

SVM demonstrated the fastest training time and achieved the highest training accuracy among the models evaluated. However, its simplicity limits its ability to classify ambiguous or complex inputs compared with DL models. By contrast, the DL models (CNN, RNN, and ANN) provided significantly better results, reaching up to 99% accuracy during both training and testing.

Overall, nearly all the algorithms yielded acceptable results on the NIST test set, confirming their suitability for HDR tasks. While the SVM had the shortest execution time, the CNN required the longest execution time.

In future work, we plan to enhance recognition accuracy by focusing entirely on deep learning techniques, exploring alternative architectures, and optimizing learning strategies. Our approach will be applied to more complex, real-world datasets, such as handwritten digits from bank checks. Furthermore, we aim to investigate adaptive segmentation methods to better handle overlapping digits and challenging handwriting styles, providing a clear roadmap for advancing offline handwritten digit recognition systems.

REFERENCES

[1] M. Islam, A. Rahman, and S. Khan, “Pre-trained CNN Models for Bengali Handwritten Digit Recognition,” arXiv preprint arXiv:2209.13005, 2022.

[2] A. Fateh, R. Singh, and P. Sharma, “Handwritten Digits Recognition Using Transfer Learning with Attention Mechanisms,” *ResearchGate*, 2024.

[3] S. Keshari, P. Jain, and R. Kumar, “Transfer Learning for Handwritten Character Classification with Limited Training Data,” *Journal of Image and Graphics*, vol. 11, no. 1, 2023. [Online]. Available: https://www.joig.net/uploadfile/2023/JOIG-V11N1-21.pdf.

[4] M. Ullah, A. Khan, S. Ali, and J. Kim, “Hybrid CNN-SVM Model for Handwritten Digit Recognition on MNIST Dataset,” arXiv preprint arXiv:2503.06104, 2025.

[5] A. Fateh, R. T. Birgani, M. Fateh, and V. Abolghasemi, “Advancing Multilingual Handwritten Numeral Recognition With Attention-Driven Transfer Learning,” *IEEE Access*, vol. 12, pp. 41381–41395, 2024, doi: 10.1109/ACCESS.2024.3245678.

[6] D. Ghosh, D. Chaurasia, S. Mondal, and A. Mahajan, “Handwritten Documents Text Recognition with Novel Pre-processing and Deep Learning,” in *Proc. 2021 Grace Hopper Celebration India (GHCI)*, IEEE, 2021, pp. 1–5, doi: 10.1109/GHCI.2021.00012.

[7] D. Gupta and S. Bag, "CNN-Based Multilingual Handwritten Numeral Recognition: A Fusion-Free Approach," *Expert Systems with Applications*, vol. 165, 113784, 2021, doi: 10.1016/j.eswa.2020.113784.

[8] B. Vidhale, G. Khekare, C. Dhule, P. Chandankhede, A. Titarmare, and M. Tayade, "Multilingual Text & Handwritten Digit Recognition and Conversion of Regional Languages into Universal Language Using Neural Networks," in *Proc. 2021 6th International Conference for Convergence in Technology (I2CT)*, IEEE, 2021, pp. 1–5, doi: 10.1109/I2CT51072.2021.9413545.

[9] A. Fateh, M. Fateh, and V. Abolghasemi, "Multilingual Handwritten Numeral Recognition Using a Robust Deep Network Joint with Transfer Learning," *Information Sciences*, vol. 581, pp. 479–494, 2021, doi: 10.1016/j.ins.2021.01.097.

[10] S. Alghyaline, "Arabic optical character recognition: A review," *Computer Modeling in Engineering & Sciences*, vol. 135, no. 3, pp. 1825–1861, 2023, doi: 10.32604/cmes.2022.024555.

[11] H. Kusetogullari, A. Yavariabdi, J. Hall, and N. Lavesson, "DIGITNET: A deep handwritten digit detection and recognition method using a new historical handwritten digit dataset," *Big Data Research*, vol. 23, p. 100182, Feb. 2021, doi: 10.1016/j.bdr.2020.100182.

[12] M. Rahman et al., "Handwritten Digit Recognition Using Feature-Based Machine Learning Approaches," arXiv preprint arXiv:2201.10102, 2022.

[13] S. Chakraborty et al., "Bengali Handwritten Digit Recognition Using Pre-trained Deep Learning Models," arXiv preprint arXiv:2209.13005, 2022.

[14] M. Islam, S. A. Shuvo, M. S. Nipun, R. B. Sulaiman, J. Nayeem, Z. Haque, M. M. Shaikh, and M. S. Ullah Sourav, "Efficient approach of using CNN based pretrained model in Bangla handwritten digit recognition," *arXiv preprint arXiv:2209.13005*, 2022.

[15] A. Choudhary and S. Ahlawat, "Hybrid CNN-SVM Classifier for Handwritten Digit Recognition," *ResearchGate*, 2025.

[16] D. O. I. Effendi, "Handwritten Digits Detection Using Convolutional Neural Network," Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI), vol. 11, no. 2, pp. 346–356, Jun. 2025.

[17] A. Kumar et al., "Hybrid CNN-SVM Approach for Handwritten Digit Recognition with Data Augmentation," arXiv preprint arXiv:2503.06104, 2025.

[18] J. Smith et al., "Ensemble Methods for Handwritten Digit Recognition: Combining CNN and Traditional ML Classifiers," arXiv preprint arXiv:2503.06105, 2025.

[19] L. Zhang et al., "Optimization of CNN Models for Handwritten Digit Recognition Using Mini-batch and Hessian-Free Techniques," arXiv preprint arXiv:2311.01022, 2023.

[20] H. Noori, "Improving Persian Handwritten Digit Recognition using Convolutional Neural Network," *Journal of Algorithms and Computation*, vol. 56, no. 1, pp. 55–72, 2024, doi: 10.22059/jac.2024.374614.1213.

[21] F. Jawad and N. R. Hamza, "Farsi Digit Recognition Using GAN-Generated Data and Convolutional Neural Networks," *Journal of Kufa for Mathematics and Computer*, vol. 12, no. 1, pp. 6–11, May 2025, doi: 10.31642/JoKMC/2018/120102.

[22] A. Nasr-Esfahani, M. Bekrani, and R. Rajabi, "Robust Persian Digit Recognition in Noisy Environments Using Hybrid CNN-BiGRU Model," *arXiv preprint arXiv:2412.10857*, Dec. 2024.

[23] A. Zohrevand, Z. Imani, J. Sadri, and C. Y. Suen, "Does color modalities affect handwriting recognition? An empirical study on Persian handwritings using convolutional neural networks," *arXiv preprint arXiv:2307.12150*, Jul. 2023.

[24] M. Bonyani, S. Jahangard, and M. Daneshmand, "Persian handwritten digit, character and word recognition using deep learning," *International Journal on Document Analysis and Recognition*, vol. 24, no. 1–2, pp. 133–143, Apr. 2021, doi: 10.1007/s10032-021-00368-2.

[25] *Machine Vision Inspection Systems: Image Processing, Concepts, Methodologies, and Applications*, dokumen.pub. [Online]. Available: https://dokumen.pub/machine-vision-inspection-systems-image-processing-concepts-methodologies-and-applications-1119681804-9781119681809.html. Accessed: Nov. 20, 2023.

[26] S. Amiri, "Image Denoising to Enhance Character Recognition Using Deep Learning," ResearchGate, 2021. [Online]. Available: https://www.researchgate.net/publication/352147349_Image_Denoising_to_enhance_Character_Recognition_using_Deep_Learning.

[27] L. Huang, J. Qin, Y. Zhou, F. Zhu, L. Liu, and L. Shao, "Normalization Techniques in Training DNNs: Methodology, Analysis and Application," *arXiv preprint arXiv:2009.12836*, 2020.

[28] A. Chakraborty, R. De, S. Malakar, F. Schwenker, et R. Sarkar, *Full segmentation system for handwritten digit string recognition*, 2021.

[29] D. Prabha Devi, R. Ramya, P. S. Dinesh, C. Palanisamy, and G. Sathish Kumar, "Design and simulation of handwritten recognition system," *Materials Today: Proceedings*, vol. 45, pp. 626–629, Jan. 2021, doi: 10.1016/j.matpr.2020.02.720.

[30] Wang Chao, "Research on Features Extraction and Classification for Images based on Transformer Learning," *Proc. 2024 International Conference on Machine Learning and Intelligent Computing*, PMLR, vol. 245, pp. 67–75, Apr. 2024.

[31] Rafal Ali Sameer, "A General Overview on the Categories of Image Features Extraction Techniques: A Survey," *J. of Al-Qadisiyah for Computer Science and Mathematics*, vol. 15, no. 1, pp. 179–188, Apr. 2023.

[32] M. Ferdous Wahid, Md. Fahim Shahriar, and Md. Shohanur Islam Sobuj, "A Classical Approach to Handcrafted Feature Extraction Techniques for Bangla Handwritten Digit Recognition," *arXiv preprint arXiv:2201.10102*, Jan. 2022.

[33] A. Adhesh Garg et al., "An Efficient CNN-ELM Classifier for Handwritten Digits Recognition and Classification," *Symmetry*, vol. 12, no. 10, Article 1742, 2020, doi: 10.3390/sym12101742.

[34] عنابت - مختار باجي جامعت, [Online]. Available: https://docplayer.fr/120345916-Jm`t-bjy-mkhtr-`nbt.html. Accessed: Nov. 20, 2023.

[35] A. Joshi, I. [surname missing], H. Pandey, and Y. R. Choudhary, "Handwritten Character Recognition by Using Machine Learning: A Review," SSRN, 2023, [Online]. Available: https://ssrn.com/abstract=4489054.

[36] *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers* [PDF]. [Online]. Available: https://www.researchgate.net/publication/277299933_Efficient_Learning_Machines_Theories_Concepts_and_Applications_for_Engineers_and_System_Designers. Accessed: Nov. 20, 2023.

[37] M. Wahid, M. M. Rahman, M. M. Hasan, et M. A. Hossain, "Handwritten Digit Recognition Using Feature-Based Machine Learning Approaches," *arXiv preprint arXiv:2201.10102*, 2022. [Online]. Available: https://arxiv.org/abs/2201.10102.

[38] W. Homenda, A. Jastrzebska, and W. Pedrycz, "Rejecting foreign elements in pattern recognition problem — reinforced training of rejection level," in *Proc. International Conference on Agents and Artificial Intelligence*, Lisbon, Portugal, 2015, pp. 90–99, doi: 10.5220/0005207900900099.

[39] S. Arya, "An optimal algorithm for approximate nearest neighbor searching in fixed dimensions."

[40] R. Karakaya and S. Kazan, "Handwritten digit recognition using machine learning," *Sakarya University Journal of Science*, vol. 25, no. 1, pp. 65–71, Feb. 2021, doi: 10.16984/saufenbilder.801684.

[41] S. Chauhan, S. Mahmood, and T. Poongodi, "Handwritten digit recognition using deep neural networks," *Proc. [Nom de la conférence]*, May 2023, pp. 1–6, doi: 10.1109/ICIEM59379.2023.10167391.

[42] S. Al-Mansoori, "Intelligent handwritten digit recognition using artificial neural network," 2015, doi: 10.13140/RG.2.1.2466.0649.

[43] J. L. Elman, "Finding structure in time," *Cognitive Science*, vol. 14, no. 2, pp. 179–211, Apr. 1990, doi: 10.1016/0364-0213(90)90002-E.

[44] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, vol. 1, D. E. Rumelhart and J. L. McClelland, Eds., Cambridge, MA, USA: MIT Press, 1988, pp. 399–421, doi: 10.1016/B978-1-4832-1446-7.50035-2.

[45] Khan, A., Sohail, A., Zahoora, U., & Qureshi, A. S. (2020). *A Survey of the Recent Architectures of Deep Convolutional Neural Networks.* Artificial Intelligence Review, 53, 5455–5516.

[46] Nalluri, S. D., Thota, S. M., & Thota, A., "Handwritten Digit Recognition Using CNN," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 11, no. 1, pp. 8849–8855, 2023.

[47] P. Agrawal *et al.*, "Automated bank check verification using image processing and deep learning methods," *Multimedia Tools and Applications*, vol. 80, no. 4, pp. 5319–5350, Feb. 2021, doi: 10.1007/s11042-020-09818-1.

[48] S. Ahlawat, A. Choudhary, A. Nayyar, S. Singh, and B. Yoon, "Improved handwritten digit recognition using convolutional neural networks (CNN)," *Sensors*, vol. 20, no. 12, p. 3344, Jun. 2020, doi: 10.3390/s20123344.

[49] "Comparisons of KNN, SVM, BP and CNN for handwritten digit recognition," ResearchGate, doi: 10.1109/AEECA49918.2020.9213482.

[50] A. Ashiquzzaman and A. K. Tushar, "Handwritten Arabic numeral recognition using deep learning neural networks," arXiv preprint arXiv:1702.04663, Feb. 15, 2017, doi: 10.48550/arXiv.1702.04663.

[51] S. Sarraf, "French word recognition through a quick survey on recurrent neural networks using long-short term memory RNN-LSTM," arXiv preprint arXiv:1804.03683, Apr. 10, 2018, doi: 10.48550/arXiv.1804.03683.

[52] M. Husnain et al., "Recognition of Urdu handwritten characters using convolutional neural network," Applied Sciences, vol. 9, no. 13, Art. no. 132758, Jan. 2019, doi: 10.3390/app9132758.

Residual DDPG Control with Error-Aware Reward Rescaling for Active Suspension Under Unseen Road Conditions

Zien Zhang[1], Abdul Hadi Abd Rahman[2]*, Noraishikin Zulkarnain[3]
Faculty of Information Science & Technology, UKM, Bangi, Selangor 43000 Malaysia[1]
Center for Artificial Intelligence Technology, UKM, Bangi, Selangor 43000 Malaysia[2]
Faculty of Engineering and Built Environment, UKM, Bangi, Selangor 43000 Malaysia[3]

***Abstract*—This study investigates a hybrid residual control framework combining Deep Deterministic Policy Gradient (DDPG) and a Proportional–Integral–Derivative (PID) based correction module for active suspension (AS) systems, aiming to improve ride performance and generalization under complex road excitations. The DDPG controller is trained on sinewave inputs, while the PID module compensates for residual errors to enhance robustness. To further guide policy optimization, an error-aware reward rescaling strategy is introduced during training, adaptively shaping the reward signal based on acceleration deviation. The controller is tested under five typical road conditions. These include sinewave inputs and step inputs, and ISO 8608 Level B random profiles. Simulation results show that the residual DDPG (RDDPG) controller works better than both DDPG alone and the PID controller. It reduces vertical acceleration RMS by 50.35% under a 0.05 m sinewave input. This shows that using reinforcement learning (RL) with fast correction and reward adjustment is a useful and stable way to control AS in different driving conditions.**

***Keywords*—Deep deterministic policy gradient; active suspension; reward function; generalization**

I. Introduction

In recent years, data-driven control methods such as RL have developed quickly. These methods have demonstrated strong adaptability and robustness in uncertain conditions [1]. At the same time, active anti-roll bar systems have shown strong potential in improving vehicle lateral stability [2]. Body roll can be reduced a lot during sharp turns or on rough roads [3], [4], [5]. These improvements show a new method to develop smart AS control. At the same time, more advanced methods are being tried in AS systems to better handle changing and unseen driving situations.

To enhance adaptability, Nguyen [6] developed a fuzzy logic PI controller that adjusts its gains in real-time, leading to improved ride comfort and stability in a quarter-car model under varying conditions. Similarly, Zhou et al. [7] introduced a machine learning model that incorporates fuzzy logic to mitigate vertical vibrations in a two degree of freedom (2-DOF) suspension setup. The method lowered body acceleration well and stayed simple to compute, so it could be used in real-time.

As the vehicle systems become more nonlinear and sophisticated, conventional controllers don't always guarantee good performance. For this reason, researchers look into neural networks (NN) and deep learning for better suspension force control. Bongain and Jamett [8] developed a hybrid fuzzy logic NN controller tailored to the nonlinear dynamics of electro-hydraulic actuators. They compared this development with the PID-ANN controller and showed that they achieved superior tracking performance in input variations. Konoiko et al. [9] introduced a feedforward deep NN trained using supervised learning. The data were acquired from the optimal PID controller by full state feedback. An automatic structure optimization procedure was performed to improve generalization.

AS control has improved a lot in recent years as more people use RL [10]. RL helps create control strategies by letting the system learn from the environment. This means it does not need a set system model. Liang and Wei [11] applied the DDPG algorithm to both active and semi-AS systems. The controller gives continuous force signals to the actuators and uses a reward function that punishes large vertical acceleration and too much suspension movement. Their results showed that the DDPG method worked well and made the ride more comfortable in different driving situations. Yong et al. [12] made a switching learning system based on the Soft Actor-Critic (SAC) framework. The system was able to recognize various types of road disturbances in real-time and adjust its control strategy accordingly. The method was first tested in simulation, then applied to real road situation. It showed that it could handle different driving conditions well.

Researchers have used actor-critic (AC) setups to better deal with uncertainty and changes in the system. Fares and Bani Younes [13] implemented a Time-Difference Advantage Actor Critic (TD-A2C) algorithm in a model-free setup. Their controller worked without knowing the environment before. It used an actor network to make actions, and a critic to check them. This setup helped keep a good balance between control force and ride comfort. Chen and Chang [14] applied a similar AC framework to control uncertain discrete-time systems. By using an NN to estimate the value function, their method kept the policy stable and made control more accurate. In parallel, Li and Kalabic [15] introduced a safety RL approach was introduced, integrating model free learning with a safety monitoring component. Penalty functions were used to reduce unsafe behavior during training which helped the controller explore effectively within safety limits.

*Corresponding Author.

Robustness and real-time adaptability remain key challenges in AS applications. To address this, Radac and Lala [16] formulated the suspension control task as a zero sum differential game between the controller and a worst case disturbance. Their game theoretic approach ensured robustness during learning by explicitly modeling adversarial conditions. Complementing this, Wang, Tian, and Zhang [17] developed a two layer control architecture based on a vehicle model. They combined incremental PID control with a deep RL adaptive (DRLA) strategy to enable online tuning of suspension forces in response to heave, pitch, and roll motions. Their method improved both ride stability and road handling. Wang, Zhuang, and Yin [18] applied DRL techniques to a quarter-car model under bump disturbances, confirming the controller's effectiveness in managing transient shocks.

The main contribution of this study is the use of error-aware reward rescaling method to make the training of the DDPG controller faster and more stable. The residual framework is combined with DDPG and PID to reduce vertical acceleration under different unseen road conditions.

II. METHODOLOGY

This study employs a classic quarter car model 2-DOF, representing the vertical motions of the sprung and unsprung masses. To guide the learning process, an Error-Aware Reward Rescaling mechanism is designed, where vertical acceleration minimization is set as the primary objective, and suspension deflection and control effort are treated as secondary considerations.

A. Quarter-Car Model and Parameters

The quarter-car model with 2-DOF represents the vertical motions of the sprung mass and the unsprung mass, respectively. The system dynamic equations and schematic diagram are given as follows, in Fig. 1:

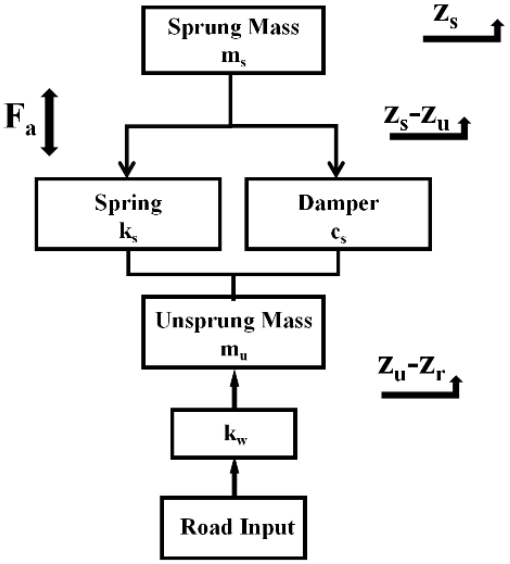


Fig. 1. Schematic diagram of the quarter-car AS system.

Sprung mass dynamics:

$$m_s\ddot{z}_s=-k_s(z_s-z_u)-c_s(\dot{z}_s-\dot{z}_u)+F_a \quad (1)$$

Unsprung mass (wheel) dynamics:

$$m_u\ddot{z}_u=-k_s(z_s-z_u)-c_s(\dot{z}_s-\dot{z}_u)-k_t(z_u-z_r)-F_a \quad (2)$$

where, m_s and m_w denote the sprung and unsprung masses, respectively; k_s and c_s represent the suspension stiffness and damping coefficient; k_w is the tire stiffness; z_s, z_u, and z_r denote the vertical displacements of the vehicle body, wheel, and road surface, respectively; and F_a is the control force generated by the AS actuator.

B. Control Framework and DDPG Controller

The control system developed in this study consists of a PID controller combined with an RL controller based on the DDPG algorithm. The DDPG controller generates the primary control force for the AS, while the PID controller learns to produce a residual correction term. The final suspension control force is the sum of the outputs from both controllers. This hybrid residual structure enhances the stability and responsiveness of the system during training as well as under unseen operating conditions. The overall control framework of the proposed AS system is illustrated in Fig. 2.

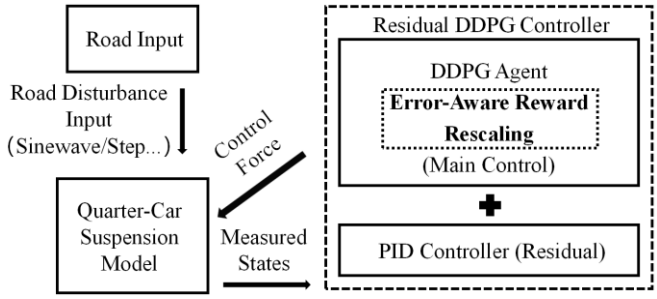


Fig. 2. Overall control framework of the AS system based on RDDPG.

The DDPG algorithm is a model-free RL method tailored for continuous control tasks. It integrates a deterministic policy gradient framework with an actor–critic architecture. The critic approximates the action-value function $Q(s, a;\theta^Q)$, which is updated by minimizing the temporal-difference loss between the estimated and target Q-values:

$$L(\theta^Q) = E[(y_t - Q(s_t, a_t; \theta^Q))^2] \quad (3)$$

The target value y_t is computed using delayed versions of the actor and critic networks:

$$y_t=r_t+\gamma\cdot Q'(s_{t+1},\mu'(s_{t+1})) \quad (4)$$

where, $\gamma\epsilon[0,1]$ is the discount factor, and μ', Q' are the target actor and critic networks respectively. These targets are held fixed for stability during mini-batch updates.

The actor represents the policy $\mu(s;\theta^\mu)$, which is updated by maximizing the critic's evaluation of the current policy. The gradient of the objective with respect to the actor's parameters is given by:

$$\nabla_{\theta^\mu}J\approx E\left[\nabla_a Q(s,a;\theta^Q)|_{a=\mu(s)}\cdot\nabla_{\theta^\mu}\mu(s)\right] \quad (5)$$

This update encourages the actor to select actions that yield higher predicted Q-values, as determined by the critic. To mitigate oscillations during learning, the algorithm employs

target networks that are updated gradually using a soft update rule:

$$\theta^{Q'} \leftarrow \tau\theta^{Q} + (1-\tau)\theta^{Q'}, \theta^{\mu'} \leftarrow \tau\theta^{Q\mu} + (1-\tau)\theta^{\mu'} \quad (6)$$

A small update coefficient τ ensures that target networks evolve slowly, thus smoothing the learning process. To promote exploration in a deterministic policy framework, temporally correlated noise is added to the actor's output:

$$a_t = \mu(s_t) + N_t \quad (7)$$

where, N_t typically follows an Ornstein–Uhlenbeck (OU) process, which models physical systems more realistically than uncorrelated noise and helps prevent premature convergence.

In this study, a residual structure is employed to improve the robustness of the DDPG controller. The RL agent is designed to fine-tune the output of the traditional controller rather than replace it entirely. The final control signal is obtained by combining the baseline controller's action with the correction learned by the agent, formulated as:

$$u_t = uPID(t) + uDDPG(t) \quad (8)$$

where, uPID denotes the output of a conventional PID controller, and uDDPG is the corrective term learned by the DDPG agent. This structure enables the RL component to focus on modeling deviations or nonlinearities that are difficult to capture with classical methods.

C. Error-Aware Reward Rescaling and Training Setup

This study introduces an Error-Aware Reward Rescaling mechanism that improves ride comfort by reducing vertical acceleration while accounting for suspension travel and control effort. The approach sets vertical acceleration as the primary objective and treats suspension deflection and actuator force as supporting factors. By adaptively shifting focus to the variable with the largest deviation at any given moment, it helps to guide the learning process more effectively, leading to better convergence and improved control reliability.

The instantaneous reward is composed of three terms: vertical comfort score $R_{comfort}$, suspension travel score R_{travel}, and control effort penalty P_{action}. The first two terms are modeled using exponential decay functions to penalize large deviations from ideal states:

$$R_{comfort} = \exp\left(-\left(\frac{|\ddot{z}_b|}{\ddot{z}_{lim}}\right)^2\right) \quad (9)$$

$$R_{travel} = \exp\left(-\left(\frac{|x_s|}{x_{lim}}\right)^2\right) \quad (10)$$

The control penalty is implemented using a hyperbolic tangent function to suppress excessively large control actions:

$$P_{action} = \tanh\left(\left(\frac{|F|}{F_{max}}\right)^2\right) \quad (11)$$

where, $\ddot{z}_b$ denotes the vertical acceleration of the vehicle body. x_s denotes the suspension travel. F denotes the actuator control force. $\ddot{z}_{lim}$ denotes the predefined upper limit for ideal vertical acceleration. x_{lim} is defined as the maximum desired suspension travel range. F_{max} denotes the maximum control force amplitude. λ refers to the control force penalty coefficient.

Instead of applying fixed weights to the comfort and travel terms, the proposed method adaptively rescales their importance based on the normalized error magnitude of each state component. Specifically, the dynamic weights are computed as:

$$w_c = \frac{\min\left(1, \frac{|\ddot{z}_b|}{\ddot{z}_{lim}}\right)}{\min\left(1, \frac{|\ddot{z}_b|}{\ddot{z}_{lim}}\right) + \min\left(1, \frac{|x_s|}{x_{lim}}\right)}, \; w_t = 1 - w_c \quad (12)$$

This formulation enables the controller to dynamically assign greater attention to the more critical performance objective at each moment, based purely on state error levels, without requiring access to prior reward outputs. As a result, the reward function guides the policy to minimize the dominant deviation and improve adaptability across varying conditions.

The final reward expression is given by:

$$r_t = w_c \cdot R_{comfort} + w_t \cdot R_{travel} - \lambda \cdot P_{action} \quad (13)$$

An additional bonus of 0.5 is granted when all normalized indicators fall below 0.5, encouraging convergence toward ideal operational zones. Numerical safety measures, including reward clipping and invalid value checks, are implemented to ensure stable learning. While the current reward function effectively guides the training process, the investigation into alternative reward shaping strategies lies outside the primary scope of this work and is reserved for future research.

The DDPG agent receives three state variables as input: suspension deflection, the corresponding vertical velocity of the sprung mass, and the vertical acceleration of the vehicle body. The output is a one-dimensional continuous control force ranging from −1000 N to 1000 N, which is applied to the AS system. Both the Actor and Critic networks adopt a single hidden layer architecture with 256 neurons. To enhance exploration, OU noise with a standard deviation of 0.5 is introduced. Detailed parameters are provided in the corresponding Table I.

TABLE I. DETAILED PARAMETERS OF DDPG CONTROLLER

DDPG Training Parameters	
Description	***Value***
Number of Hidden Neurons	256
Learning Rate (Actor & Critic)	0.001
Target Network Soft Update Factor	0.001
Exploration Noise Type	OU
Noise Standard Deviation	0.5
Experience Replay Buffer Size	1e6

The control input of the AS system is the suspension force generated by the RDDPG controller. This control structure consists of two components: a primary controller based on DDPG, which is trained under smooth road conditions, and a residual correction module based on a PID controller that compensates for the DDPG output. The controller receives system state variables as input, including the vertical acceleration of the vehicle body. The DDPG module outputs an initial control force, while the PID module learns the residual

component to refine the control output. This hybrid architecture enhances adaptability and robustness under previously unseen scenarios.

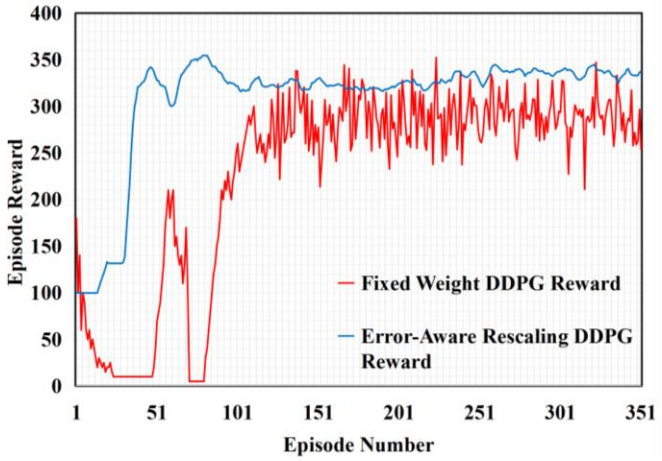


Fig. 3. Episode reward of Error-Aware Rescaling DDPG and Fixed Weight DDPG.

As shown in the training reward curve in Fig. 3, the DDPG agent utilizing the proposed Error-Aware Rescaling reward function demonstrates significantly superior learning performance compared to the baseline with fixed-weight rewards. Specifically, the rescaling-based agent achieves rapid convergence within the first 50 episodes and maintains consistently higher episode rewards with minimal fluctuations throughout the training process. In contrast, the fixed-weight reward agent exhibits unstable learning behavior, with prolonged periods of near-zero rewards and considerable oscillations even in later stages. This indicates that the fixed-weight design is prone to suboptimal policy learning and lacks robustness. Overall, the Error-Aware Rescaling approach enhances both convergence speed and policy stability, leading to improved final performance and demonstrating strong potential for guiding RL in suspension control tasks.

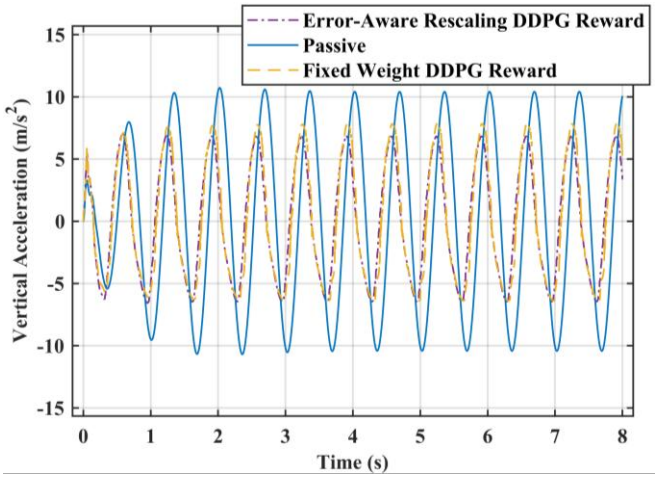


Fig. 4. Comparison of Error Aware Rescaling DDPG and Fixed Weight DDPG.

During training, a sinewave road excitation with a frequency of 1.5 Hz and amplitudes of 0.05m is used to simulate typical smooth driving conditions. The DDPG controller is trained until the policy converges, as indicated by a stable reward trend and a continuous reduction in the root mean square (RMS) of vertical acceleration.

TABLE II. SIMULATION RESULTS OF ERROR-AWARE RESCALING DDPG AND FIXED-WEIGHT DDPG

Controller	Compare DDPG at Sinewave 0.05m Road Input Condition	
	RMS	***RMS Reduction***
Passive	7.11 m/s²	-
Error-Aware Rescaling DDPG	4.62 m/s²	35.02%
Fixed-Weight DDPG	5.05 m/s²	28.94%

The results are shown in Fig. 4 and Table II. The passive suspension system produces an RMS value of 7.11 m/s², indicating significant vibration and poor ride comfort. When using the DDPG controller trained with the proposed error-aware rescaling reward, the RMS is reduced to 4.62 m/s², achieving a 35.02% improvement over the passive case. In comparison, the controller trained with fixed reward weights yields an RMS of 5.05 m/s², corresponding to a 28.94% reduction. The improved reward structure demonstrates a more pronounced effect in enhancing the vehicle's vertical stability and ride comfort.

III. SIMULATION SETUP

To evaluate the control performance and generalization capability of the proposed RDDPG control strategy in AS systems, this study conducts simulation research based on a quarter-car suspension model. The entire simulation environment is established within the MATLAB/Simulink platform. Both the training and testing procedures are carried out using a representative 2-DOF quarter AS model, which comprises the sprung mass, unsprung mass, suspension spring, damper, and active actuator. The system dynamics take into account the vertical motions of the vehicle body and wheel, aiming to characterize the suspension's response under various road excitations. Table III presents the detailed parameters of the vehicle.

TABLE III. DETAILED PARAMETERS OF VEHICLE

Vehicle Parameters	
Value	***Description***
1400 kg	Sprung mass (full car)
45 kg	Unsprung mass
25800 N/m	Suspension spring stiffness
1200Ns/m	Suspension damping coefficient
198000N/m	Tire stiffness

To evaluate the generalization capability of the proposed controller, three types of unseen road excitation profiles are introduced during the testing phase. The first type is a step road excitation, which includes step inputs of 0.03 m (unseen) and 0.05 m (trained) in amplitude. This test is designed to simulate sudden road height changes, such as speed bumps or curbs. The corresponding profile is illustrated in Fig. 5.

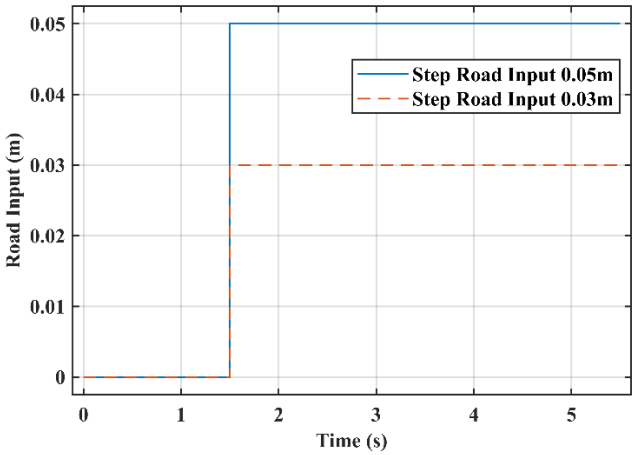


Fig. 5. Step 0.03m and step 0.05m road input profiles.

The second type is a random road excitation, constructed according to the ISO 8608 standard for road roughness classification. Specifically, Level B profiles are employed to represent medium and poor road conditions, respectively. These excitations are shown in Fig. 6.

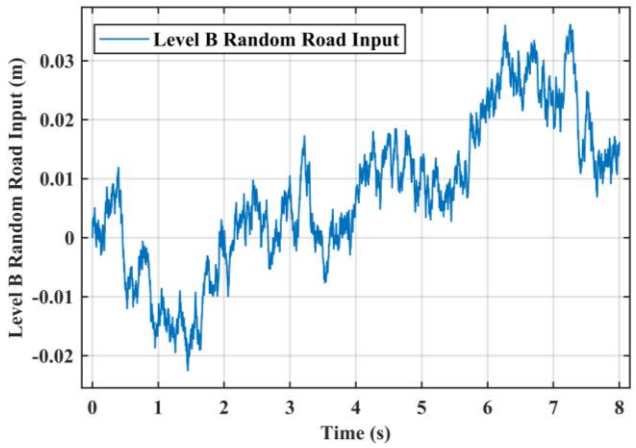


Fig. 6. Level B random road input profiles.

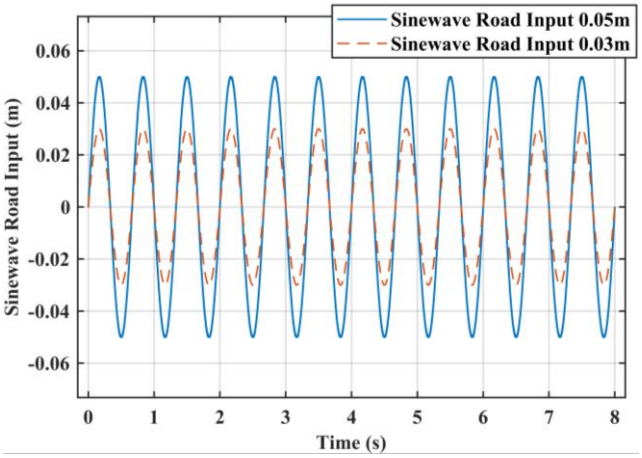


Fig. 7. Sinewave 0.05m and Sinewave 0.03m road input profiles.

The third type is a sinewave excitation, which is identical to the road profile used in the training phase. This sinewave 0.05m scenario serves as a baseline to validate the controller's fundamental performance under familiar conditions. This sinewave 0.03m is an unseen condition to validate the controller's performance. The excitation waveform is presented in Fig. 7.

IV. Simulation Results

To comprehensively assess the effectiveness and generalization ability of the proposed RDDPG control strategy under varying road conditions, three representative test scenarios were constructed: sinewave 0.05m input excitation (training condition), sinewave 0.03m input excitation (unseen condition), step 0.03m and 0.05m input (unseen condition), and ISO 8608 Level B random road profiles (unseen conditions). The figures present the vertical acceleration responses of the vehicle body under different control strategies, including passive suspension, PID control, standalone DDPG control, and the proposed RDDPG controller, when subjected to these road excitations.

A. Sinewave 0.05m Road Input Condition Simulation Results

The simulation results under the sinewave 0.05m road input condition are shown below:

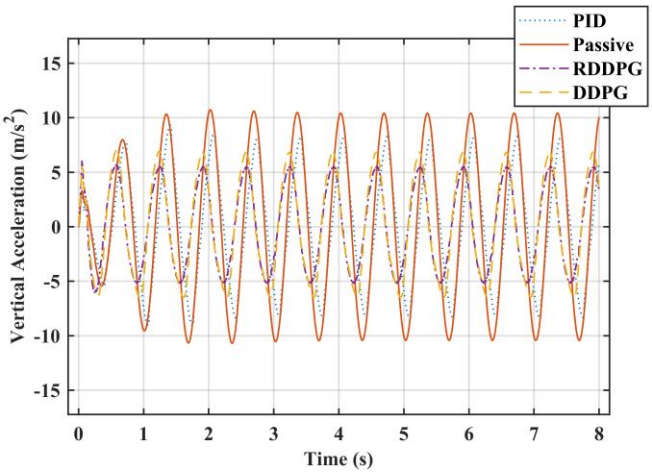


Fig. 8. Sinewave 0.05m road input simulation results.

During the training phase, a sinewave road excitation with an amplitude of 0.05 m was employed. Fig. 8 illustrates the vertical acceleration responses of the vehicle body under different control strategies. As shown, all three active control methods significantly reduce the fluctuation amplitude of vertical acceleration compared to the passive suspension, with the proposed RDDPG controller delivering the most pronounced improvement. According to the data summarized in Table IV, the passive suspension system RMS vertical acceleration of 7.11 m/s². The DDPG-based controller reduces this value to 4.62 m/s², corresponding to a 35.02% decrease, while the conventional PID controller achieves an RMS of 5.24 m/s², reflecting a 26.30% reduction. Notably, the RDDPG approach further lowers the RMS to 3.53 m/s², achieving a 50.35% reduction over the passive baseline and demonstrating the best overall performance in vibration attenuation. These results confirm the effectiveness and superiority of the RDDPG strategy under 0.05m sinewave excitation conditions.

TABLE IV. SIMULATION RESULTS OF SINEWAVE 0.05M ROAD INPUT CONDITION

Controller	Sinewave 0.05m Road Input Condition	
	RMS	***RMS Reduction***
Passive	7.11 m/s²	-
DDPG	4.62 m/s²	35.02%
RDDPG	3.53 m/s²	50.35%
PID	5.24 m/s²	26.30%

B. Sinewave 0.03m Road Input Condition Simulation Results

The simulation results under the sinewave 0.03m road input condition are shown below:

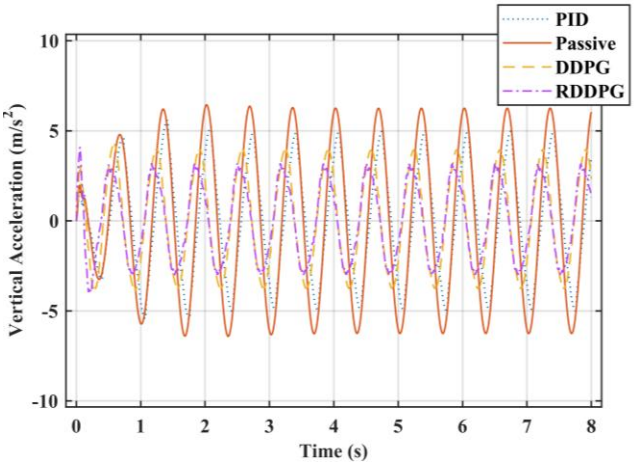


Fig. 9. Sinewave 0.03m road input simulation results.

To assess the vibration isolation performance under moderate road disturbances, a sinewave input with an amplitude of 0.03 m was applied in the simulation. The corresponding vertical acceleration responses for each control method are presented in Fig. 9. As observed, all AS strategies provide improved performance over the passive system, with the RDDPG controller demonstrating the most notable attenuation. According to the results summarized in Table V, the passive suspension yields an RMS vertical acceleration of 4.26 m/s². The DDPG controller reduces this to 2.84 m/s², indicating a 33.33% improvement, while the PID controller achieves 3.14 m/s², corresponding to a 26.29% reduction. The RDDPG strategy achieves the lowest RMS value of 2.14 m/s², marking a 49.77% decrease relative to the passive configuration. Overall, the results highlight the RDDPG controller's robust adaptability and its superior vibration suppression performance under reduced excitation amplitudes.

TABLE V. SIMULATION RESULTS OF SINEWAVE 0.03M ROAD INPUT CONDITION

Controller	Sinewave 0.03m Road Input Condition	
	RMS	***RMS Reduction***
Passive	4.26 m/s²	-
DDPG	2.84 m/s²	33.33%
RDDPG	2.14 m/s²	49.77%
PID	3.14 m/s²	26.29%

C. Step 0.05m Road Input Condition Simulation Results

The simulation results under the step 0.05m road input condition are shown below:

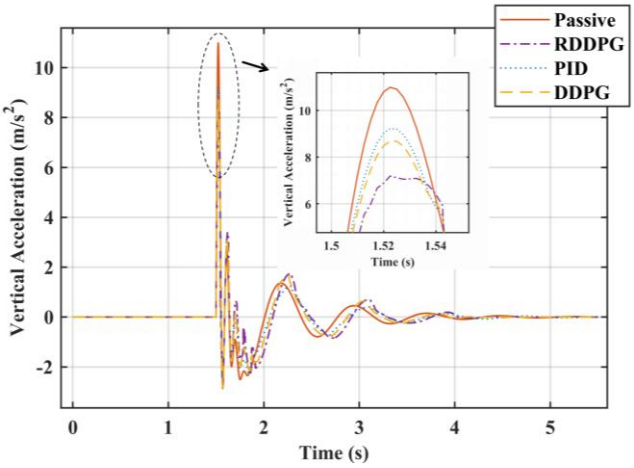


Fig. 10. Step 0.05m road input simulation results.

To evaluate the generalization performance of each control strategy under previously unseen conditions, a step road disturbance with a height of 0.05 m was introduced, simulating an abrupt change in road profile. The resulting vertical acceleration responses are illustrated in Fig. 10. Compared to the passive suspension, all three active control strategies clearly mitigate the initial impact and help the system stabilize more rapidly. The magnified section highlights the effectiveness of each controller in suppressing peak acceleration. The corresponding RMS values are listed in Table VI. The passive system yields an RMS vertical acceleration of 0.99 m/s². The DDPG controller reduces it to 0.82 m/s², resulting in a 17.17% decrease. The PID controller achieves 0.84 m/s², reducing the RMS by 15.15%. The RDDPG controller attains the best result with 0.79 m/s², representing a 20.20% reduction compared to the passive baseline. These findings demonstrate that the RDDPG strategy not only performs well under trained periodic conditions but also retains strong robustness and adaptability when facing sudden road disturbances.

TABLE VI. SIMULATION RESULTS OF STEP 0.05M ROAD INPUT CONDITION

Controller	Step 0.05m Road Input Condition	
	RMS	***RMS Reduction***
Passive	0.99 m/s²	-
DDPG	0.82 m/s²	17.17%
RDDPG	0.79 m/s²	20.20%
PID	0.84 m/s²	15.15%

D. Step 0.03m Road Input Condition Simulation Results

The simulation results under the step 0.03m road input condition shown below:

Fig. 11 illustrates the vertical acceleration responses of the vehicle body under different control strategies. Compared to the passive suspension, the active controllers effectively

suppress the initial impact peak and accelerate the system's convergence. The magnified section in the figure provides a detailed comparison of the overshoot behavior across controllers. Table VII summarizes the corresponding RMS values. The passive system yields an RMS vertical acceleration of 0.58 m/s². The DDPG controller reduces this to 0.48 m/s², representing a 15.79% improvement. The PID controller achieves 0.50 m/s², which corresponds to a 13.79% reduction. The RDDPG controller achieves the lowest RMS at 0.45 m/s², yielding a 21.05% decrease compared to the passive baseline. These results further demonstrate that the proposed RDDPG control strategy maintains reliable performance and strong generalization capability even under mild, abrupt disturbances.

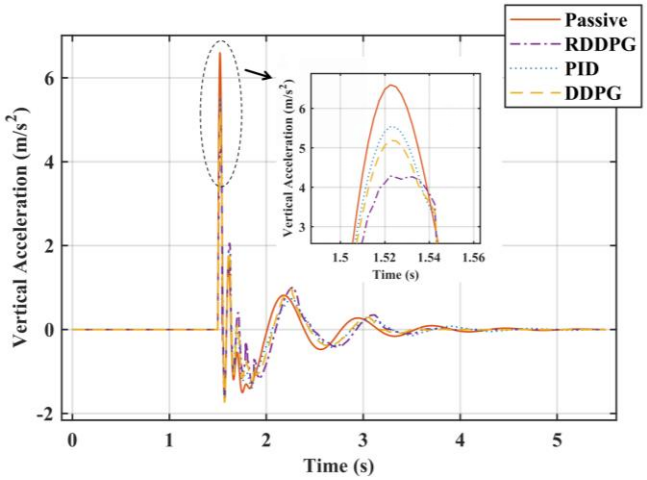


Fig. 11. Step 0.03m road input simulation results.

TABLE VII. SIMULATION RESULTS OF STEP 0.03M ROAD INPUT CONDITION

Controller	Step 0.03m Road Input Condition	
	RMS	***RMS Reduction***
Passive	0.58 m/s²	-
DDPG	0.48 m/s²	15.79%
RDDPG	0.45 m/s²	21.05%
PID	0.50 m/s²	13.79%

E. Level B Road Input Condition Simulation Results

The simulation results under the ISO 8608 Level B random road input condition are shown below:

The ISO 8608 Class B random road profile was introduced to examine how well each control strategy adapts to irregular and complex surface disturbances. Fig. 12 displays the resulting vertical acceleration responses. As shown, the passive suspension experiences pronounced high-frequency oscillations, whereas the active control strategies noticeably suppress these fluctuations, indicating their improved ride comfort capabilities. Table VIII reports the corresponding RMS values. The passive configuration results in an RMS of 0.76 m/s². The DDPG strategy improves this to 0.63 m/s², yielding a 17.11% reduction, while the PID controller achieves 0.66 m/s², reflecting a 13.16% decrease. RDDPG again provides the best result, further lowering the RMS to 0.62 m/s², an 18.42% improvement over the passive baseline. These findings suggest that the RDDPG approach retains a high level of performance even under stochastic mid-grade excitations, demonstrating greater resilience and adaptability than the other methods tested.

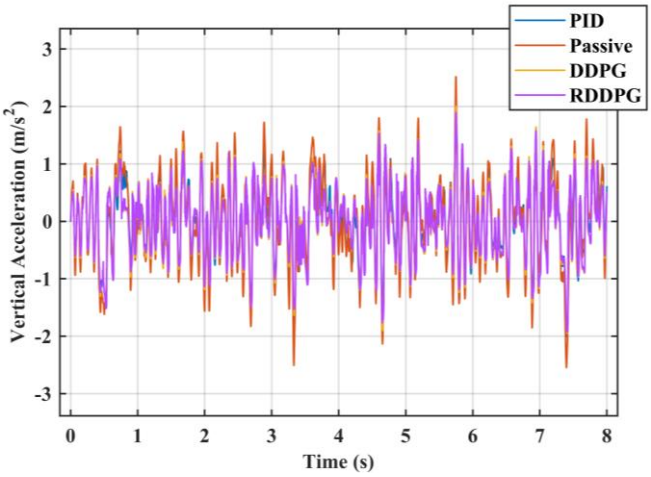


Fig. 12. Level B random road input simulation results.

TABLE VIII. SIMULATION RESULTS OF LEVEL B RANDOM ROAD INPUT CONDITION

Controller	Level B Random Road Input Condition	
	RMS	***RMS Reduction***
Passive	0.76 m/s²	-
DDPG	0.63 m/s²	17.11%
RDDPG	0.62 m/s²	18.42%
PID	0.66 m/s²	13.16%

V. DISCUSSION

Different from the residual control structure proposed by Hynes et al. [19], this study adopts an RL-centric architecture, where the DDPG controller serves as the primary decision-making module, and a PID controller is employed as a residual correction mechanism to enhance robustness. In contrast to their design, which relies on a PID-based policy and uses DDPG as a supplementary component requiring careful pre-tuning, this study approach eliminates the dependence on a well-tuned initial controller. Moreover, an Error-Aware Reward Rescaling mechanism is introduced to improve both training efficiency and policy convergence. The prior work reports results under a limited range of conditions. In contrast, this study tests the proposed method across five representative road scenarios, including both training and unseen cases, to provide a more complete assessment of its generalization and robustness.

Compared to other RL suspension control studies [20], this study demonstrates clear advantages in control architecture, adaptability, and generalization capability. First, it uses a residual control setup that combines a DDPG controller with a PID correction model. The PID component adaptively adjusts the DDPG output to enhance the stability and robustness of the control strategy under complex road conditions. An Error-Aware Reward Rescaling mechanism is introduced to adjust

reward weights based on acceleration errors during training, which improves adaptability and convergence efficiency. In terms of validation, this study evaluates the controller under five representative road profiles that include sinewaves, step disturbances, and ISO 8608 Level B random inputs. This study provides a more comprehensive assessment of generalization compared to previous work.

Simulation results consistently show that RDDPG outperforms standalone DDPG and traditional PID controllers. Under the 0.05 m sinewave excitation used as the training condition, RDDPG achieves a 50.35% reduction in RMS vertical acceleration compared to the passive baseline, exceeding the performance of DDPG at 35.02% and PID at 26.30%. Under the 0.05 m step input as an unseen condition, RDDPG still delivers the best result, reducing RMS by 20.20%, while DDPG achieves 17.17% and PID achieves 15.15%. Even in stochastic and unstructured conditions such as ISO 8608 Level B random road condition, RDDPG maintains reliable performance with the lowest RMS value of 0.62 m/s², indicating strong resilience.

Overall, by combining the long-term policy optimization ability of RL with the short-term corrective precision of PID control, the RDDPG strategy achieves a balanced performance in both stability and generalization. Its consistent effectiveness across a wide range of scenarios underscores its reliability and practical value for AS systems.

VI. Conclusion

This study adopts a hybrid control strategy for AS systems based on a residual DDPG framework. The method uses a DDPG controller together with a PID controller in a residual setup. The PID part adjusts the DDPG output to make the system more stable and robust. To further improve training efficiency and convergence, an Error-Aware Reward Rescaling strategy is introduced during the learning process. The mechanism adjusts the reward signal based on vertical acceleration deviation to guide the agent toward better control behavior. The proposed method is tested under five typical road excitations, including both training scenarios and several unseen cases. This work makes the method better by adding a way to change the reward during training. This assists to keep training stable and improves control. The results show that the better RDDPG works well in both tested and new situations, and gives a useful way to control active suspension.

VII. Future Work

Future work will try using the RDDPG method on full-vehicle suspension models to test how well it works with more axles and connected movements. Additionally, incorporating real-time implementation and validation will be considered to assess the controller's practical feasibility and robustness in real-world scenarios. Further research on alternative residual structures and adaptive weighting mechanisms may also enhance the system more flexible and work better.

Acknowledgment

This research was funded by the Ministry of Higher Education Malaysia and Universiti Kebangsaan Malaysia through FRGS research grants (Vote no: FRGS/1/2022/TK02/UKM/02/2 and a university research grant (Vote no. GGPM-2017-087), respectively.

References

[1] M. Raibail et al., "Decentralized Multi-Robot Collision Avoidance: A Systematic Review from 2015 to 2021," Symmetry, vol. 14, no. 3, p. 610, Mar. 2022, doi: 10.3390/sym14030610.

[2] N. Zulkarnain, H. Zamzuri, and S. A. Mazlan, "LQG Control Design for Vehicle Active Anti-Roll Bar System," Appl. Mech. Mater., vol. 663, pp. 146–151, Oct. 2014, doi: 10.4028/www.scientific.net/AMM.663.146.

[3] N. Binti Zulkarnain et al., "Newly Developed Nonlinear Vehicle Model for an Active Anti-roll Bar System," Bull. Electr. Eng. Inform., vol. 7, no. 4, pp. 529–537, Dec. 2018, doi: 10.11591/eei.v7i4.1185.

[4] N. Zulkarnain, A. Hussain, H. Zamzuri, and S. S. Mokri, "Controller Design for an Active Anti-roll Bar System," in 2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA), Xi'an, China: IEEE, Jun. 2019, pp. 2213–2217. doi: 10.1109/ICIEA.2019.8834106.

[5] Z. Zhang, A. H. A. Rahman, N. Zulkarnain, and T. Zhang, "Enhancing Vehicle Lateral Stability: A DDPG-Based Active Anti-Roll Bar Control Strategy," IEEE Access, vol. 12, pp. 153030–153043, 2024, doi: 10.1109/ACCESS.2024.3480116.

[6] D. N. Nguyen and T. A. Nguyen, "A Hybrid Control Algorithm Fuzzy-PI with the Second Derivative of the Error Signal for an Active Suspension System," Math. Probl. Eng., vol. 2022, pp. 1–10, Aug. 2022, doi: 10.1155/2022/3525609.

[7] H. Zhou, V. Nguyen, R. Jiao, and Y. Huan, "Application of machine learning method to control the vibration of the car's suspension system," Vibroengineering PROCEDIA, vol. 38, pp. 44–49, Jun. 2021, doi: 10.21595/vp.2021.22025.

[8] S. Bongain and M. Jamett, "Electrohydraulic Active Suspension Fuzzy-Neural Based Control System," IEEE Lat. Am. Trans., vol. 16, no. 9, pp. 2454–2459, Sep. 2018, doi: 10.1109/TLA.2018.8789568.

[9] A. Konoiko, A. Kadhem, I. Saiful, N. Ghorbanian, Y. Zweiri, and M. N. Sahinkaya, "Deep learning framework for controlling an active suspension system," J. Vib. Control, vol. 25, no. 17, pp. 2316–2329, Sep. 2019, doi: 10.1177/1077546319853070.

[10] S. B. Samsuri, H. Zamzuri, M. A. A. Rahman, S. A. Mazlan, and A. H. A. Rahman, "Computational Cost Analysis of Extended Kalman Filter in Simultaneous Localization & Mapping (EKF-Slam) Problem for Autonomous Vehicle," vol. 10, no. 17, 2015.

[11] G. Liang, T. Zhao, and Y. Wei, "DDPG based self-learning active and model-constrained semi-active suspension control," in 2021 5th CAA International Conference on Vehicular Control and Intelligence (CVCI), Tianjin, China: IEEE, Oct. 2021, pp. 1–6. doi: 10.1109/CVCI54083.2021.9661158.

[12] H. Yong, J. Seo, J. Kim, M. Kim, and J. Choi, "Suspension Control Strategies Using Switched Soft Actor-Critic Models for Real Roads," IEEE Trans. Ind. Electron., vol. 70, no. 1, pp. 824–832, Jan. 2023, doi: 10.1109/TIE.2022.3153805.

[13] A. Fares and A. Bani Younes, "Online Reinforcement Learning-Based Control of an Active Suspension System Using the Actor Critic Approach," Appl. Sci., vol. 10, no. 22, p. 8060, Nov. 2020, doi: 10.3390/app10228060.

[14] H.-C. Chen, Y.-C. Lin, and Y.-H. Chang, "An Actor-Critic Reinforcement Learning Control Approach for Discrete-Time Linear System with Uncertainty," in 2018 International Automatic Control Conference (CACS), Taoyuan: IEEE, Nov. 2018, pp. 1–5. doi: 10.1109/CACS.2018.8606740.

[15] Z. Li, T. Chu, and U. Kalabic, "Dynamics-Enabled Safe Deep Reinforcement Learning: Case Study on Active Suspension Control," in 2019 IEEE Conference on Control Technology and Applications (CCTA), Hong Kong, China: IEEE, Aug. 2019, pp. 585–591. doi: 10.1109/CCTA.2019.8920696.

[16] M.-B. Radac and T. Lala, "Learning nonlinear robust control as a data-driven zero-sum two-player game for an active suspension system,"

IFAC-Pap., vol. 53, no. 2, pp. 8057–8062, 2020, doi: 10.1016/j.ifacol.2020.12.2243.

[17] W. Wang, K. Tian, and J. Zhang, "Dynamic Modelling and Adaptive Control of Automobile Active Suspension System," J. Eur. Systèmes Autom., vol. 53, no. 2, pp. 297–303, May 2020, doi: 10.18280/jesa.530218.

[18] X. Wang, W. Zhuang, and G. Yin, "Learning-Based Vibration Control of Vehicle Active Suspension," in 2020 IEEE 18th International Conference on Industrial Informatics (INDIN), Warwick, United Kingdom: IEEE, Jul. 2020, pp. 94–99. doi: 10.1109/INDIN45582.2020.9442091.

[19] A. Hynes, E. Sapozhnikova, and I. Dusparic, "Optimising PID Control with Residual Policy Reinforcement Learning," presented at the 28th Irish Conference on Artificial Intelligence and Cognitive Science, Dec. 2020.

[20] A. Ahmad, M. A. Radi, A. Ahmad, and A. Boudiaf, "Reinforcement Learning-Based Optimal Control Design for Active Suspension Systems," in 2024 Advances in Science and Engineering Technology International Conferences (ASET), Abu Dhabi, United Arab Emirates: IEEE, Jun. 2024, pp. 1–6. doi: 10.1109/ASET60340.2024.10708702.

Edge-Guided Multi-Scale YOLOv11n: An Advanced Framework for Accurate Ship Detection in Remote Sensing Imagery

Yan Shibo[1*], Liu Pan[2], Abudhahir Buhari[3]
Faculty of Engineering, Science and Technology, Kuala Lumpur University of Science and Technology, Kuala Lumpur, Malaysia[1, 2, 3]
Department of Information Technology, Anhui Vocational College of Defense Technology, Lu'an, China[1]

***Abstract*—Ship detection in optical remote sensing imagery plays a vital role in maritime surveillance and environmental monitoring. However, existing deep learning models often struggle to generalize effectively in complex marine environments due to challenges such as noise interference, small object sizes, and diverse weather conditions. To address these issues, this study proposes an Edge-Guided Multi-Scale YOLO algorithm (YOLOv11n-EGM). The approach introduces multi-scale deep convolutional branches with varying kernel sizes to perform parallel feature extraction, enhancing the model's ability to detect objects of different scales. Additionally, the classic Sobel operator is incorporated for edge-aware feature extraction, improving the model's sensitivity to object boundaries. Finally, 1×1 convolutions are employed for feature fusion, reducing computational complexity. Experimental results on the ShipRSImageNet V1.0 dataset demonstrate that the improved model achieves notable gains in precision, recall, mAP@0.5, and mAP@0.5:0.95 compared to the baseline, highlighting its superior performance in challenging maritime scenarios. Qualitative analysis further shows that YOLOv11n-EGM can accurately detect both large and extremely small ships in cluttered scenes, with precise boundary localization. However, occasional misclassification in fine-grained categories (e.g., motorboat vs. hovercraft) highlights the challenge of small-instance recognition. Overall, the proposed method exhibits strong robustness and practical applicability in real-world maritime scenarios, offering a promising solution for edge-aware, multi-scale ship detection in remote sensing imagery.**

***Keywords*—Optical remote sensing imagery; ship detection; multi-scale deep convolution; edge-aware feature extraction**

I. INTRODUCTION

Maritime transportation plays a crucial role in global trade, accounting for over 80% of total international trade volume. According to the International Maritime Organization (IMO), there are approximately 108,789 commercial vessels worldwide as of 2024. As the core carrier of marine economic activities, the operational efficiency of ships directly impacts the stability of international logistics systems [1]. Ship detection in optical remote sensing imagery has emerged as a critical research area due to its wide-ranging applications in maritime surveillance, port management, and marine traffic control. The task, commonly referred to as optical remote sensing ship detection, involves the automatic localization and classification of ship targets within high-resolution optical satellite images, and serves as a foundational technique for intelligent maritime monitoring systems [2].

However, the unique characteristics of ship targets in nadir-view remote sensing imagery—such as significant scale variation, complex backgrounds, and prominent edge structures—have posed substantial challenges to detection models. In recent years, models such as R-CNN, Fast R-CNN, and the YOLO series have gained attention for their effectiveness in visual target identification [2], [3], [4]. Among them, the YOLO series has demonstrated particular advantages in optical remote sensing applications due to its high detection speed and competitive accuracy [2].

Nevertheless, despite the general success of YOLO-based detectors in object detection tasks, their direct application to ship detection in optical remote sensing imagery—e.g., YOLOv11n—reveals several domain-specific limitations. These include difficulty in handling ships of drastically varying sizes, insufficient exploitation of edge and contour features critical for distinguishing ships from complex backgrounds, and degraded performance in fine-grained classification tasks such as differentiating 49 ship types and docks. YOLOv11n's baseline architecture lacks specialized modules to address the unique challenges posed by maritime scenes [2]. These limitations indicate a clear research gap in designing lightweight yet accurate detectors that can effectively capture multi-scale and edge-sensitive features in complex maritime scenes.

To address this gap, this study proposes an enhanced YOLOv11-based approach with the following key contributions:

A novel multi-scale deep convolutional module, EGM_Block, is introduced, employing parallel convolutional kernels of sizes 3, 5, and 7 to capture features at multiple scales.

1) A Sobel edge extraction module is incorporated to improve the model's sensitivity to edge features of ship targets.

2) An efficient feature fusion strategy is proposed to reduce computational complexity while maintaining high detection accuracy.

3) Extensive experiments are conducted on the publicly available ShipRSImageNet V1.0 dataset, which provides a comprehensive benchmark for evaluating ship detection performance in remote sensing imagery. The results validate the effectiveness and robustness of the proposed method.

The remainder of this study is organized as follows: Section II reviews related work; Section III details the proposed method; Section IV presents experimental results and analysis; Section V provides a discussion of the findings; and Section VI concludes the study and discusses future work.

II. Related Work

A. Traditional Methods for Ship Detection

Early approaches to ship detection in remote sensing imagery primarily relied on handcrafted features and classical machine learning algorithms. Techniques such as edge detection, texture analysis, and morphological operations were commonly employed to extract candidate regions, followed by classifiers like Support Vector Machines (SVM) or AdaBoost for final decision-making. While these methods offered a degree of interpretability, they were limited in their ability to handle variations in scale, orientation, and complex backgrounds, resulting in suboptimal performance in real-world maritime environments [2].

B. Deep Learning-Based Ship Detection

The emergence of deep learning has significantly advanced the field of object detection. Two-stage detectors, such as R-CNN, Faster R-CNN [3], and Mask R-CNN [4], introduced region proposal mechanisms that substantially improved detection accuracy, particularly for small and densely packed objects. After thorough experimental analysis and engineering-level testing, the two-stage model proved highly effective for ship detection in remote sensing imagery. However, in practical engineering applications, the two-stage model is difficult to apply due to real-time or resource-constrained environments, such as satellite-attached equipment or unmanned aerial vehicles [5].

To address the aforementioned challenges, researchers have developed single-stage detectors, such as SSD [6] and the YOLO ("You Only Look Once") series [7], etc. The single-stage detection architecture eliminates the step of region proposal and directly predicts bounding boxes and class probabilities. This design achieves a good balance between speed and accuracy. The YOLO series has been widely used in industry applications due to its real-time performance and continuous architectural improvements [8].

C. Evolution of the YOLO Series

With the joint efforts of researchers and engineers, the YOLO series has undergone multiple efficient iterations in a short period of time. Strictly speaking, each new version surpasses its predecessor in terms of accuracy, speed, and feature representation. YOLOv3 introduced multi-scale detection and residual connections [9], while YOLOv4 [10] and YOLOv5 [11] adopted techniques such as CSPNet, PANet, and data augmentation strategies. YOLOv5 has shown practical effectiveness in agricultural applications, including crop detection and maturity classification under real-world conditions [12]. Recent versions, including YOLOv7 and YOLOv8, have focused on optimizing the backbone network and introducing lightweight variants suitable for edge deployment [13].

YOLOv11 is one of the latest advancements in the YOLO object detection series. Not only has its backbone architecture been optimized, but its feature extraction modules have also been enhanced, resulting in improved detection accuracy and computational efficiency [14] [15]. Therefore, YOLOv11 generally performs better than previous versions in various general object detection tasks, and it is especially suitable for scenarios requiring real-time inference.

Nonetheless, when applied to ship detection in high-resolution optical remote sensing images, there are still some domain-specific challenges. Compared with conventional datasets, optical remote sensing images are captured from different viewing angles, resulting in large variations in ship scales, orientations, and arrangement densities. In particular, ships may appear in various environments such as ports, coastal areas, and open seas. These background conditions lead to significant changes in the overall appearance and spatial distribution of the images, which may reduce the effectiveness of standard detection procedures.

In addition, experimental results show that YOLO models may struggle to capture fine edge details and the contextual cues necessary to distinguish ships from background elements such as waves, docks, and infrastructure. Therefore, accurately identifying ships that are slender, small in size, and densely arranged in images remains a key challenge. Moreover, in real-world engineering scenarios where resources are limited and inference speed is critical—such as onboard or satellite systems—balancing detection accuracy and computational efficiency is also of great importance [16].

Considering the above-mentioned practical technical difficulties, there is a clear need for specialized architectural improvements that can address issues such as multi-scale representation, edge-aware feature learning, and stable performance in complex environments. The YOLOv11n-EGM framework proposed in this study aims to tackle these challenges by integrating an edge-guided multi-scale module and an adaptive loss strategy, thereby enhancing the reliability of ship detection in remote sensing imagery.

D. Challenges in Remote Sensing Ship Detection

Remote sensing imagery presents unique challenges for object detection algorithms. With the growing application of remote sensing in coastal environment monitoring, recent studies have extended its practical relevance to broader maritime scenarios [17]. In complex maritime environments, such as freezing seas, remote sensing systems may produce erroneous object detection results without proper surface discrimination mechanisms [18]. These images are captured from nearly vertical viewing angles, which often results in limited visible features of objects. Additionally, due to the large coverage area, ships frequently appear in various sizes and orientations within the same image, and are often situated in complex environmental conditions, such as cluttered port

backgrounds, cloud cover, sea surface noise, and low contrast [19]. Moreover, ships typically exhibit distinct edge features that can be leveraged to improve detection accuracy; however, many existing models fail to fully utilize this information. Furthermore, for applications in real-time maritime monitoring systems, lightweight and computationally efficient models are essential [20] [21].

III. METHODOLOGY

This study introduces a lightweight ship detection framework specifically designed for optical remote sensing imagery—Edge-Guided Multi-Scale YOLO. The framework integrates multi-scale perception and edge contour enhancement mechanisms. Its architecture is based on an improved YOLOv11n backbone and is further enhanced by a custom-designed Edge-Guided Multi-Scale block (EGM-block), which improves the model's ability to capture structural and edge-related features across different scales.

A. Overall Network Architecture

The proposed method adopts YOLOv11n as the base detection framework and integrates the Edge-Guided Multi-Scale block (EGM-block) into the backbone network to enhance feature extraction capabilities. The backbone consists of multiple convolutional and residual layers, which progressively capture spatial and semantic representations from the input image. The EGM-Block is inserted at the fourth stage (P3/8) to enhance mid-level features in terms of edge awareness and multi-scale object perception. The C3k2 and C2PSA modules further refine deeper semantic features, while the SPPF module contributes to multi-scale contextual aggregation.

B. Edge-Guided Multi-Scale Block (EGM-Block)

As illustrated in Fig. 1, the proposed EGM-Block is composed of three functional components designed to enhance the model's ability to capture structural and scale-aware features in remote sensing imagery.

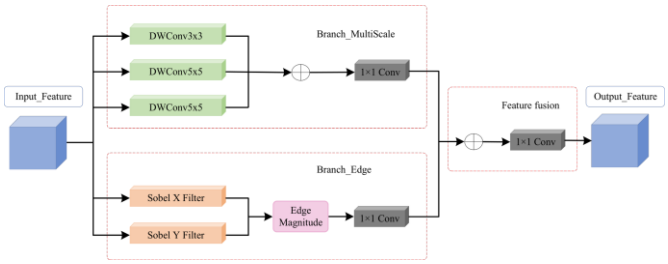


Fig. 1. Edge-Guided Multi-Scale Block.

1) Multi-scale depthwise convolutional branches: This component utilizes parallel depthwise separable convolutions with fixed kernel sizes of 3×3, 5×5, and 7×7 to extract features across multiple spatial resolutions. Such a design enables the model to effectively perceive ships of varying sizes, thereby improving its adaptability to scale diversity in optical remote sensing scenes.

2) Sobel-based edge enhancement branch: To reinforce contour sensitivity, fixed-weight Sobel filters are employed to extract directional edge information. Specifically, horizontal and vertical edge responses are obtained by convolving the input feature map with predefined Sobel kernels along the x and y axes. These directional responses are subsequently fused through a 1×1 convolution to generate an edge-enhanced feature representation. Here, $edge_x$ and $edge_y$ denote the horizontal and vertical edge responses, respectively, while edge represents the overall edge magnitude. A small constant varepsilon = 1×10^{-6} is added to ensure numerical stability during square root computation. The overall edge magnitude is then computed using the Euclidean norm, as formulated in Eq. (1):

$$edge = \sqrt{edge_x^2 + edge_y^2 + 1 \times 10^{-6}} \quad (1)$$

3) Feature fusion module: The outputs from the multi-scale and edge branches are concatenated along the channel dimension and integrated using a 1×1 convolution to produce the final enhanced representation.

C. Attention Mechanism and Feature Enhancement

To improve the model's ability to focus on informative regions, the C2PSA (Parallel Spatial Attention) module—originally integrated into the YOLO11 architecture—is retained in the upper layers of the backbone. This attention mechanism operates by applying parallel spatial attention across feature maps, enabling the network to suppress irrelevant background noise while enhancing the response of target-relevant regions. Although not proposed in this work, the inclusion of C2PSA contributes to improved detection accuracy, particularly in complex remote sensing environments such as ports and coastal zones.

D. Loss Function and Training Strategy

To address the challenges of complex maritime environments, YOLOv11n-EGM incorporates a tailored loss weighting strategy. This approach assigns greater importance to overlapping ship instances, enhancing the model's responsiveness to occluded and densely packed targets, particularly in port areas. The edge-guided multi-scale enhancement mechanism further contributes to precise boundary localization. When combined with regression loss, it effectively reduces false detections caused by cluttered backgrounds.

The training process adheres to the standard YOLOv11n framework, utilizing the Stochastic Gradient Descent (SGD) optimizer with an initial learning rate of 0.01, adjusted dynamically through cosine annealing. While the core training methodology remains consistent, the integration of the EGM-Block and the customized loss scheme significantly improves the model's robustness and generalization capabilities. Detailed implementation settings, including data augmentation techniques and hardware configurations, are provided in Section IV.

E. Implementation Details

The proposed model is implemented using the PyTorch deep learning framework, leveraging its modular design and GPU acceleration capabilities. Input images are uniformly resized to 640×640 pixels to maintain consistency across

training batches and to balance computational efficiency with spatial resolution. The model is trained with a batch size of 64, which allows for stable gradient updates while fully utilizing the memory capacity of high-performance GPUs.

To address the substantial scale variation of ship targets—ranging from small fishing boats to large cargo vessels—the architecture incorporates an Edge-Guided Multi-Scale (EGM) feature enhancement strategy. This mechanism enables the network to capture fine-grained details and contextual cues across different receptive fields, thereby improving detection accuracy for both small and large objects. In addition, the EGM-Block, embedded within the backbone, helps suppress irrelevant background features such as waves, docks, and coastal infrastructure, which often lead to false alarms in remote sensing imagery.

All experiments are conducted on the publicly available ShipRSImageNet V1.0 dataset, which provides a comprehensive benchmark for ship detection. The dataset includes 3,435 high-resolution images annotated with horizontal bounding boxes (HBB), oriented bounding boxes (OBB), and polygon masks. It is designed with a hierarchical classification structure across four task levels (Level 0 to Level 3), covering 50 distinct ship categories [5]. A detailed overview of the dataset and training configuration is presented in Section IV.

IV. Experimental Results and Analysis

To validate the effectiveness of the proposed YOLOv11n-EGM framework, a series of experiments is conducted on the ShipRSImageNet V1.0 dataset. This section presents the dataset characteristics, training configuration, evaluation metrics, and comparative results against baseline models. The experiments are designed to assess the model's performance across varying task complexities and ship categories.

A. Dataset Overview

This study utilizes the ShipRSImageNet V1.0 dataset, a large-scale, fine-grained benchmark designed for ship detection in high-resolution optical remote sensing imagery. The dataset comprises 3,435 images sourced from diverse satellite platforms and sensors, spanning various geographic regions and weather conditions. Each image measures approximately 930×930 pixels and contains ships of varying scales, orientations, and aspect ratios. A total of 17,573 ship instances are annotated using horizontal bounding boxes (HBB), oriented bounding boxes (OBB), and polygon masks, offering rich spatial and contextual information.

ShipRSImageNet V1.0 introduces a hierarchical classification structure across four levels (Level 0 to Level 3), encompassing 50 distinct ship types. The updated version includes a standardized test set to facilitate fair benchmarking and reproducibility in ship detection research [5].

B. Experimental Setup and Evaluation Metrics

All models are implemented using the PyTorch framework. Input images are resized to 640×640 pixels, with a batch size of 64. Training is conducted over 300 epochs on an NVIDIA RTX 4090 GPU with 24GB of memory. Data augmentation techniques such as random rotation and scale variation are applied to enhance generalization.

To assess detection performance, the following metrics are employed:

1) mAP@0.5 and mAP@0.5:0.95: Mean Average Precision at fixed and varying IoU thresholds.

2) HBB mAP: Detection accuracy using horizontal bounding boxes.

3) Precision, Recall, and F1 Score: Standard metrics for evaluating classification and localization quality.

Performance is further analyzed across different object sizes (small, medium, large) and classification levels.

C. Comparative Methods and Baseline Models

To rigorously assess the effectiveness of the proposed YOLOv11n-EGM architecture, a series of representative object detection frameworks are selected as baselines. These models encompass diverse backbone networks and detection paradigms. Comparative experiments are conducted across all four hierarchical levels (Level 0 to Level 3) defined in the ShipRSImageNet dataset, enabling evaluation from coarse-grained classification (ship vs. non-ship) to fine-grained ship type recognition.

D. Visual Results and Quantitative Analysis

To validate the effectiveness of the proposed YOLOv11n-EGM model, we conducted comparative experiments against several mainstream object detectors. All models were trained under identical conditions to ensure a fair and unbiased evaluation. The visual and quantitative results of these comparisons are summarized in Table I.

TABLE I. Experimental Comparison Results of Different Algorithms

Task Level	Method	$mAP_{50:95}$	mAP_{50}	mAP_{75}
Level 0: 1 ships and dock	Faster R-CNN	0.432	0.674	0.487
	Deformable DETR	0.505	0.763	0.596
	YOLOv9	0.447	0.662	0.523
	YOLOv10	0.594	0.760	0.656
	YOLOv11n-EGM	**0. 606**	**0. 775**	**0.691**
Level 1: 3 ship and dock	Faster R-CNN	0.372	0.566	0.421
	Deformable DETR	0.431	0.609	0.512
	YOLOv9	0.404	0.551	0.455
	YOLOv10	0.496	0.625	0.551
	YOLOv11n-EGM	**0.519**	**0.649**	**0.606**
Level 2: 24 ships and dock	Faster R-CNN	0.368	0.527	0.421
	Deformable DETR	0.509	0.635	0.583
	YOLOv9	0.399	0.497	0.464
	YOLOv10	0.505	0.608	0.557
	YOLOv11n-EGM	**0. 511**	**0. 618**	**0.565**
Level 3: 49 ships and dock	Faster R-CNN	0.381	0.529	0.438
	Deformable DETR	0.585	0.701	**0.760**
	YOLOv9	0.433	0.525	0.493
	YOLOv10	0.584	0.668	0.650
	YOLOv11n-EGM	**0.610**	**0.703**	0.656

1) Overall performance trends: YOLOv11n-EGM consistently demonstrates superior performance across all Task Levels and evaluation metrics. Notably, in Level 0 and Level 3 scenarios, its results in mAP(50:95) and mAP_75 significantly outperform other methods, validating the effectiveness of the proposed EGM-Block in enhancing multi-scale and edge-level feature representation. YOLOv10 ranks closely behind, indicating that the YOLO series achieves a commendable balance between model efficiency and detection accuracy. In contrast, Faster R-CNN and YOLOv9 exhibit noticeable performance degradation in more complex scenes (Level 2 and Level 3), suggesting limited adaptability to dense object distributions and occlusions.

Notably, we observed different comparative results of YOLOv11n-EGM across Level 0 to Level 3 of the ShipRSImageNet V1.0 dataset. This discrepancy is primarily due to the substantial variation in classification granularity across the hierarchical levels. Specifically, Level 0 distinguishes whether an object is a ship or not, involving only two classes. Level 1 further categorizes ship objects into four broad categories. Level 2 refines these categories into 24 subcategories, while Level 3 provides the most detailed classification, identifying 49 specific ship types and one dock class, resulting in a total of 50 categories. As the classification becomes increasingly fine-grained from Level 0 to Level 3, the complexity of the detection task escalates, placing greater demands on the network's feature extraction and discrimination capabilities. Consequently, the performance of the proposed algorithm varies across levels, reflecting its sensitivity to the granularity of the classification task.

To further illustrate the detection performance of the proposed YOLOv11n-EGM model under varying task complexities, Precision-Recall (PR) curves are plotted for Level 0 and Level 3 scenarios. These curves provide a comprehensive view of the trade-off between precision and recall across different confidence thresholds.

As shown in Fig. 2, YOLOv11n-EGM achieves a mean average precision (mAP) of 0.772 at an IoU threshold of 0.5. The model maintains high precision and recall in Level 0, where the task involves binary classification (ship vs. non-ship). This indicates strong performance in relatively simple detection scenarios with clear object boundaries and low category granularity. The category-wise analysis further shows that Dock achieves an average precision of 0.797, outperforming Ship at 0.747, likely due to the more consistent structural features of docks.

In contrast, Fig. 3 presents results from the more challenging Level 3 setting, where the model is required to distinguish among 49 ship types and docks, totaling 50 fine-grained categories. The overall mAP drops to 0.702, and the PR curves exhibit greater dispersion across classes. Despite this, the model preserves a balanced PR profile, demonstrating robustness in dense and cluttered maritime environments. The performance degradation reflects the increased difficulty posed by fine-grained classification, small object sizes, and complex backgrounds. These findings reinforce the importance of edge-aware feature extraction and multi-scale representation in enhancing detection accuracy under varying conditions.

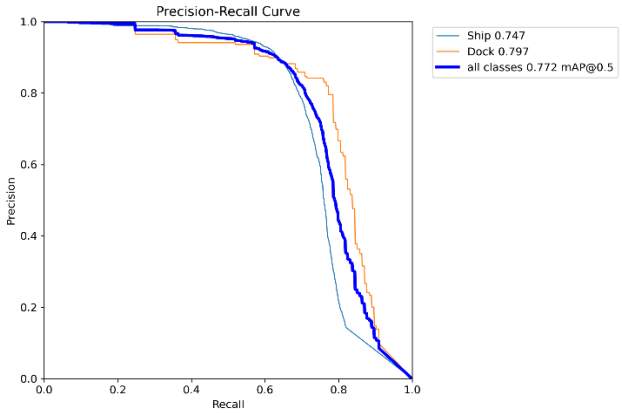


Fig. 2. Precision-Recall curve of YOLOv11n-EGM on Level 0 (lightweight scenario).

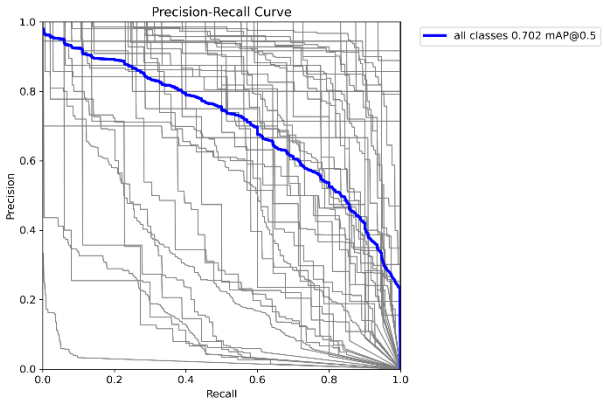


Fig. 3. Precision-Recall curve of YOLOv11n-EGM on Level 3 (complex scenario).

2) Performance variation across task levels: As the Task Level increases—corresponding to a higher number of ship and dock targets—all models experience a general decline in detection performance. However, YOLOv11n-EGM shows the least performance drop, highlighting its robustness in handling complex and cluttered environments. Specifically, in Level 3 (49 ship and dock), YOLOv11n-EGM achieves a mAP(50:95) of 0.610, surpassing both Deformable DETR (0.585) and YOLOv10 (0.584), thereby reinforcing its advantage in high-density scenarios.

To evaluate the classification accuracy across multiple ship categories, the normalized confusion matrix for Level 3 is presented in Fig. 4. This visualization highlights the model's ability to distinguish between fine-grained ship types under high-density conditions. The diagonal dominance in the matrix indicates strong classification performance, while off-diagonal elements reveal occasional misclassifications, which are primarily observed among visually similar ship classes.

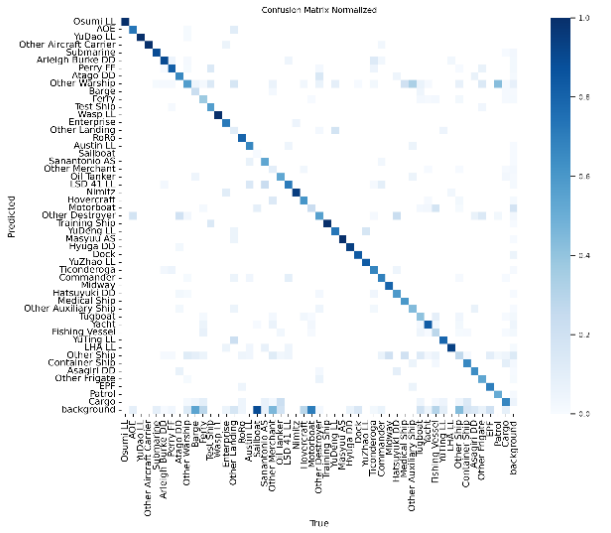


Fig. 4. Normalized confusion matrix of YOLOv11n-EGM on Level 3, illustrating multi-class classification accuracy.

The overall training dynamics of YOLOv11n-EGM are depicted in Fig. 5, which includes the evolution of key metrics such as loss, mAP, precision, and recall over 300 epochs. The smooth convergence and consistent improvement across metrics validate the stability and effectiveness of the training strategy.

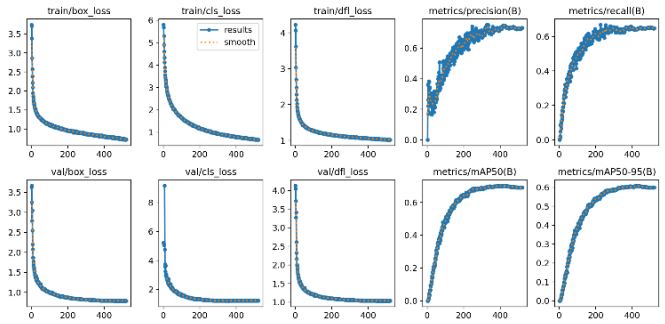


Fig. 5. Training performance of YOLOv11n-EGM over 300 epochs, including loss, mAP, precision, and recall curves.

To qualitatively assess the detection capability of YOLOv11n-EGM, Fig. 6 and Fig. 7 present sample predictions and corresponding ground truth annotations from the Level 3 validation set. The model accurately localizes and classifies multiple ship instances, even in densely packed scenes, demonstrating its practical applicability in real-world remote sensing tasks.

Notably, in several examples, both large vessels and extremely small ships appear within the same image and are successfully detected, with no significant omission. The bounding boxes closely align with the actual contours of the ships, indicating that the model effectively captures edge features and adapts to multi-scale object variations. These observations validate the effectiveness of the edge-guided multi-scale module and confirm that YOLOv11n-EGM achieves its intended improvements in fine-grained maritime detection.

However, we also observed a case where a very small ship instance (Motorboat) was misclassified as a Hovercraft. This confusion likely stems from the limited pixel-level detail available for extremely small targets, which poses a challenge for fine-grained classification.

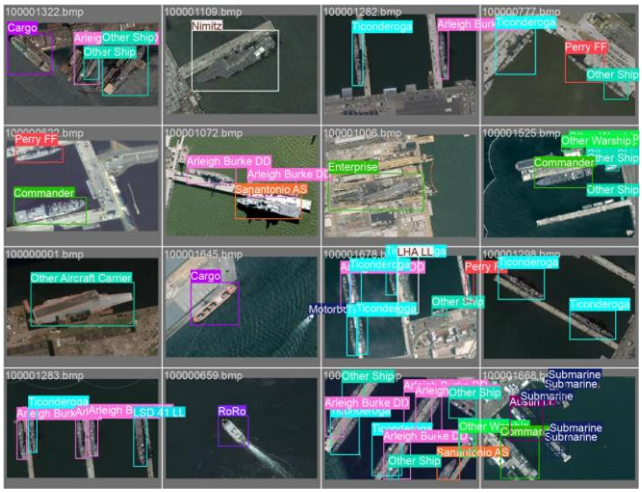


Fig. 6. Ground truth annotations on a Level 3 validation image.

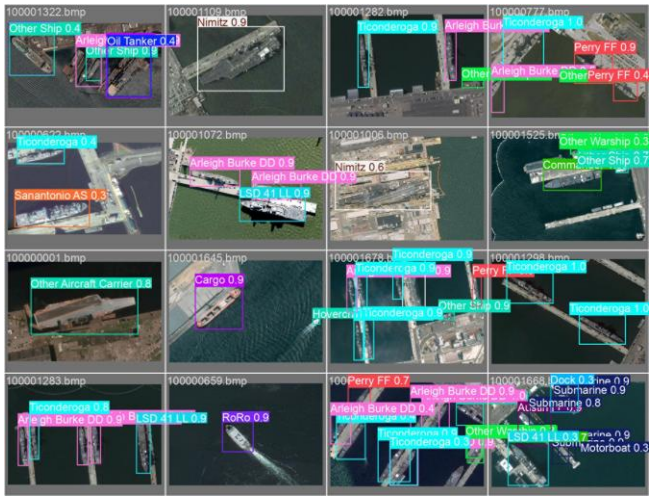


Fig. 7. YOLOv11n-EGM predictions on the same Level 3 validation image

V. Discussion

The YOLOv11n-EGM model demonstrates strong performance in ship detection tasks, particularly in cluttered optical remote sensing scenes. However, we observed that very small vessels, while successfully detected, are sometimes misclassified as other ship types. This issue arises from insufficient fine-grained semantic representation, which limits the model's ability to distinguish subtle intra-class variations. Addressing this challenge may require enhancing class-specific feature discrimination or introducing adaptive refinement mechanisms. Overall, the results affirm the potential of lightweight architectures for real-time maritime monitoring, though further improvements are needed for precise recognition of small and visually ambiguous targets.

VI. CONCLUSION

This study presents Edge-Guided Multi-Scale YOLOv11n (YOLOv11n-EGM), an enhanced object detection framework tailored for ship detection in high-resolution remote sensing imagery. Built upon the YOLOv11n backbone, the proposed model integrates a novel EGM-Block, which simultaneously strengthens multi-scale feature extraction and edge-aware representation.

Extensive experiments were conducted on the ShipRSImageNet V1.0 dataset, which includes a hierarchical structure of four task levels (Level 0 to Level 3), encompassing a wide range of ship types and scene complexities. The results demonstrate that YOLOv11n-EGM consistently outperforms several state-of-the-art detectors across all evaluation metrics, particularly in challenging scenarios with dense and small-scale targets.

The proposed EGM-Block significantly improves the model's ability to detect fine-grained ship instances while maintaining robustness against background interference. Comparative studies further confirm the critical role of the multi-scale and edge-guided branches in enhancing detection accuracy and generalization.

A. Limitations

1) Sensitivity to small instances: Qualitative analysis reveals that the model struggles to detect extremely small ships (e.g., < 8 pixels in size), often resulting in lower recall rates. This limitation is particularly critical in high-resolution satellite imagery, where small vessels are common.

2) Spectral and weather dependency: The current model relies exclusively on optical imagery, making it vulnerable to adverse weather conditions such as dense cloud cover, fog, or nighttime scenarios. This spectral dependency limits its applicability in all-weather or low-visibility environments.

3) Computational overhead: The introduction of the edge-guided multi-scale module increases the model's complexity in terms of both parameter count and GFLOPs. This added computational burden may hinder real-time deployment on edge devices or UAVs with limited processing power.

B. Future Work

1) Enhancing small object detection: We aim to improve the detection of small ships by incorporating super-resolution pre-processing pipelines and exploring anchor-free dense prediction frameworks (e.g., keypoint-based detectors). These approaches may enhance the model's ability to localize and classify tiny instances more effectively.

2) SAR-Optical data fusion: To overcome the limitations of optical-only input, we plan to integrate Synthetic Aperture Radar (SAR) data with optical imagery. SAR's all-weather, day-and-night imaging capabilities can complement optical data, enabling more robust detection under challenging environmental conditions.

3) Model compression and acceleration: To reduce computational overhead, we will explore lightweight alternatives such as depthwise separable convolutions, Ghost modules, and neural architecture search (NAS). Additionally, knowledge distillation techniques will be investigated to transfer knowledge from the full model to a compact student model suitable for real-time inference.

REFERENCES

[1] UNCTAD, Review of Maritime Transport 2024. United Nations, 2024. [Online]. Available: https://unctad.org/system/files/official-document/rmt2024_en.pdf

[2] T. Zhao *et al.*, "Ship Detection with Deep Learning in Optical Remote-Sensing Images: A Survey of Challenges and Advances," *Remote Sensing*, vol. 16, no. 7, Art. no. 7, Jan. 2024, doi: 10.3390/rs16071145.

[3] S. Wei, H. Chen, X. Zhu, and H. Zhang, "Ship detection in remote sensing image based on faster R-CNN with dilated convolution," presented at the 2020 39th Chinese control conference (CCC), IEEE, 2020, pp. 7148–7153.

[4] X. Nie, M. Duan, H. Ding, B. Hu, and E. K. Wong, "Attention mask R-CNN for ship detection and segmentation from remote sensing images," Ieee Access, vol. 8, pp. 9325–9334, 2020.

[5] Z. Zhang, L. Zhang, Y. Wang, P. Feng, and R. He, "ShipRSImageNet: A Large-Scale Fine-Grained Dataset for Ship Detection in High-Resolution Optical Remote Sensing Images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 8458–8472, 2021, doi: 10.1109/JSTARS.2021.3104230.

[6] G. Wen *et al.*, "MS-SSD: multi-scale single shot detector for ship detection in remote sensing images," *Appl Intell*, vol. 53, no. 2, pp. 1586–1604, Jan. 2023, doi: 10.1007/s10489-022-03549-6.

[7] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," presented at the Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779–788.

[8] Q. Xu, Y. Li, and Z. Shi, "LMO-YOLO: A ship detection model for low-resolution optical satellite imagery," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 15, pp. 4117–4131, 2022.

[9] L. Chen, W. Shi, and D. Deng, "Improved YOLOv3 Based on Attention Mechanism for Fast and Accurate Ship Detection in Optical Remote Sensing Images," *Remote Sensing*, vol. 13, no. 4, Art. no. 4, Jan. 2021, doi: 10.3390/rs13040660.

[10] E. Yildirim and T. Kavzoglu, "Ship Detection in Optical Remote Sensing Images Using YOLOv4 and Tiny YOLOv4," in *Innovations in Smart Cities Applications Volume 5*, M. Ben Ahmed, A. A. Boudhir, İ. R. Karaș, V. Jain, and S. Mellouli, Eds., Cham: Springer International Publishing, 2022, pp. 913–924. doi: 10.1007/978-3-030-94191-8_74.

[11] X. M. Sun, Y. J. Zhang, H. Wang, and Y. X. Du, "Research on ship detection of optical remote sensing image based on Yolo V5," *J. Phys.: Conf. Ser.*, vol. 2215, no. 1, p. 012027, Feb. 2022, doi: 10.1088/1742-6596/2215/1/012027.

[12] V. Moya, A. Quito, A. Pilco, J. P. Vásconez, and C. Vargas, "Crop detection and maturity classification using a yolov5-based image analysis," *Emerg. Sci. J.*, vol. 8, no. 2, pp. 496–512, 2024.

[13] D. Chen, H. Zhao, Y. Li, Z. Zhang, and K. Zhang, "PMF-YOLOv8: Enhanced Ship Detection Model in Remote Sensing Images," Information Technology and Control, vol. 53, no. 4, Art. no. 4, Dec. 2024, doi: 10.5755/j01.itc.53.4.37003.

[14] L. He, Y. Zhou, L. Liu, W. Cao, and J. Ma, "Research on object detection and recognition in remote sensing images based on YOLOv11," Scientific Reports, vol. 15, no. 1, p. 14032, 2025.

[15] R. Khanam and M. Hussain, "YOLOv11: An Overview of the Key Architectural Enhancements," Oct. 23, 2024, *arXiv*: arXiv:2410.17725. doi: 10.48550/arXiv.2410.17725.

[16] W. Chen, B. Han, Z. Yang, and X. Gao, "MSSDet: Multi-Scale Ship-Detection Framework in Optical Remote-Sensing Images and New Benchmark," *Remote Sensing*, vol. 14, no. 21, Art. no. 21, Jan. 2022, doi: 10.3390/rs14215460.

[17] K. M. A. Aziz, "Quantitative monitoring of coastal Erosion and changes using remote sensing in a mediterranean Delta," *Civil Eng J*, vol. 10, no. 6, pp. 1842–1862, 2024.

[18] A. Nekrasov, A. Khachaturian, and C. Fidge, "Using semicircular sampling to increase sea water/ice discrimination altitude," *Emerging Science Journal*, vol. 8, no. 2, pp. 480–495, 2024.

[19] Q. Zhang, L. Wang, H. Meng, Z. Zhang, and C. Yang, "Ship Detection in Maritime Scenes under Adverse Weather Conditions," *Remote Sensing*, vol. 16, no. 9, Art. no. 9, Jan. 2024, doi: 10.3390/rs16091567.

[20] C. Lang, X. Yu, and X. Rong, "LSDNet: a lightweight ship detection network with improved YOLOv7," *Journal of Real-Time Image Processing*, vol. 21, no. 2, p. 60, 2024.

[21] P. Tian and Z. and Li, "YOLOv8-SP: ship object detection in optical remote sensing imagery," *Marine Geodesy*, vol. 0, no. 0, pp. 1–22, doi: 10.1080/01490419.2025.2493860.

The ECTLC-Horcrux Protocol for Decentralized Biometric-Based Self-Sovereign Identity with Time-Lapse Encryption

N. M. Kaziyeva[1], R. M. Ospanov[2], N. Issayev[3], K. Maulenov[4], Shakhmaran Seilov[5]
Faculty of Information Technology, Eurasian National University, Astana, Kazakhstan[1, 2, 3, 5]
AKHMET BAITURSYNULY Kostanay Regional University, Kostanay Kazakhstan[4]

***Abstract*—In the era of rapid development of digital communication, there is a growing need for technologies that guarantee secure user identification, document authentication and protection of personal data, including biometrics. Previously used centralized identity management systems are becoming increasingly vulnerable to hacking, falsification and misuse. This problem is especially relevant when information must remain closed until a specific moment or event occurs, for example, in the fields of forensics, healthcare or law (medical certificates, legal acts, inheritance agreements, etc.). The main goal is to create a secure, verifiable and at the same time distributed access control system with the ability to defer disclosure of information. The study proposes a cryptographic protocol that combines Self-Sovereign Identity (SSI), Time-Lapse Cryptography (TLC), and decentralized biometric data management. The protocol is based on the principles of Time-Lapse Cryptography (TLC) and the Horcrux protocol, which enable time-controlled disclosure of encrypted information associated with a user's identity. The architecture includes the use of QR codes as a transport for Verifiable Credentials (VC), blockchain for authenticity verification and key management, and biometrics as a second factor of identity binding. The proposed solution is intended for use in scenarios where cryptographic protection against premature access to sensitive data is required, such as in medicine, forensics, notarial acts, or intellectual property. The study presents the protocol structure and application options.**

***Keywords*—Self-sovereign identity; horcrux protocol; elliptic curves time-lapse cryptography; biometrics; QR codes; blockchain**

I. Introduction

With the rapid growth of digital interactions, technologies that ensure reliable identification of individuals, document authentication, and protection of sensitive personal data, including biometric information, are becoming increasingly important. Traditional centralized identity management systems are increasingly vulnerable to compromise, counterfeiting, and unauthorized access. This is especially critical in cases where data cannot be disclosed before a certain event or time, such as in forensic, medical, or legal practice (medical reports, legal documents, testaments, etc.). The main challenge is to ensure secure, verifiable, and at the same time decentralized access management with the ability to delay disclosure of information.

One promising concept that can provide a balance between reliable authentication and privacy protection is the Self-Sovereign Identity (SSI) model. It is based on the idea that a user should have full ownership of their digital identities and control over the process of their provision without the participation of centralized certifying parties. SSI uses decentralized identifiers (DIDs), cryptographically protected identities (Verifiable Credentials, VC) and blockchain as a trusted, immutable infrastructure [1-3]. However, in its basic form, SSI does not solve the problem of temporary access control and secure management of biometric information.

This work is aimed at developing a new cryptographic protocol that expands the capabilities of SSI technology by combining QR codes as a transport mechanism for distributing links to VCs and DIDs, biometrics as a factor in linking credentials to a physical identity, time-lapse encryption as a mechanism for delayed access to sensitive data, and blockchain as a decentralized platform for trust and automatic key management. The key components of the proposed solution are the Horcrux protocol [4] and the Elliptic Curves Time-Lapse Cryptography (ECTLC) protocol [5]. The Horcrux protocol is a protocol which implements cryptographic separation and multi-party management of biometric templates. It eliminates the need for centralized storage of biometrics and allows the user to safely participate in the authentication process by restoring the template only with the participation of trusted parties. The Elliptic Curves Time-Lapse Cryptography (ECTLC) protocol is a cryptographic mechanism that allows data to be encrypted with the impossibility of decrypting it before a specified time, without dependence on a trusted time server. In this study, ECTLC is used to create secure VCs with delayed activity. In addition, the architecture takes into account compatibility with the Biometric Open Protocol Standard (BOPS) [6], which defines a secure way to collect, store and transmit biometric data. Together, these mechanisms are combined into a single protocol that implements biometric SSI verification with time-based access restriction, which allows for scenarios with cryptographically secured delayed disclosure of information.

The proposed solution allows for the implementation of application scenarios in which access to credentials and sensitive data is possible only after a certain point of time. It is applicable in medicine, law, education, intellectual property and other areas where strict compliance with deadlines and provable protection of identity and data is required (e.g., delayed disclosure of wills and court documents; restriction of access to biometric data activated only by a legal or time-based software trigger; time capsules and future statements

confirmed by VC and DID). Thus, such a solution makes it possible to form the basis for building trusted digital ecosystems in which the user receives technological means for managing not only identity, but also the time of access to their data, including biometrics, taking into account modern standards and decentralized protocols.

In general, the approach under consideration for the integration of SSI, TLC, QR codes, blockchain and biometrics technologies is due to the need to increase the level of trust in digital transactions, minimize the risks of personal data leakage and form identification systems that are resistant to external influences. Modern trends in the digitalization of various spheres of society (public administration, medicine, the financial sector, education, etc.) demonstrate the limitations of traditional centralized authentication mechanisms based on logins, passwords or unified registries. These models are characterized by a high degree of vulnerability to cyberattacks, leaks and misuse of information. In contrast, the concept of self-sovereign identification assumes decentralized management of personal attributes, in which the subject retains control over their own digital data and selectively discloses it depending on the context of interaction.

Using an encryption mechanism for a specified time allows expanding the functionality of identification systems, providing access to encrypted information strictly at a predetermined moment. Even if encrypted data is compromised, temporary blocking ensures that early access is impossible.

The use of QR codes provides a universal and technologically accessible channel for transmitting encrypted tokens, keys or digital certificates. This tool facilitates the coupling of physical and digital processes, allowing for the implementation of identity verification scenarios in offline environments with an instant transition to secure online interaction.

The use of biometric methods strengthens the trust basis of the system, as they allow for reliable verification that the presented digital identity actually belongs to a physical carrier. Unlike traditional authentication factors that are subject to transfer or theft, biometric features have a high degree of uniqueness and stability, which reduces the likelihood of subject substitution.

Blockchain technology guarantees the integrity and verifiability of data. This eliminates the possibility of falsifying transactions or making retroactive changes to records. The distributed structure of the registry eliminates the dependence on a centralized verification authority and eliminates the possibility of system failure. In addition, the use of smart contracts allows for automatic management of access to time-dependent resources, complementing cryptographic methods with the functionality of predefined rule execution.

As a result, SSI ensures decentralization and user control, biometrics confirm the authenticity of the individual, QR codes create a convenient interaction interface, encryption for a specified time regulates the moment of access to data, blockchain guarantees the immutability of records and automation of rules. The result is a multi-level system that combines convenience for the end user and a high level of cryptographic protection. The prospects for such integration are most significant in contexts that require a combination of confidentiality, time synchronization, and reliable identity verification.

It can be said that there are currently no works that consider the implementation of all the above components in one formal protocol at the same time. More common are works that cover two to three elements and architectural compositions that can be combined (see Table I).

TABLE I COMPARATIVE TABLE OF THE WORKS DEVOTED TO INTEGRATION OF SSI, TLC, QR CODES, BLOCKCHAIN AND BIOMETRICS TECHNOLOGIES. NOTATION: + CONSIDERED, - NOT CONSIDERED

Sources	SSI	Blockchain	Biometrics	QR-codes	TLC
[4]	+	+	+		
[7]	+	+			
[8]	+		+		
[9]	+	+	+		
[10]	+	+	+	+	
[11]	+			+	
This study	+	+	+	+	+

A. Motivation and Problem Statement

SSI improves user control and verifiability but does not guarantee cryptographic non-access to sensitive attributes before a policy time (T). Biometric authentication often relies on centralized custody. We address both gaps by combining decentralized biometric sharding with time-lapse encryption anchored to SSI.

B. Main Contributions

SSI-native protocol that binds VCs to a user via Horcrux-style biometric splitting without centralized template storage;

ECC time-lapse layer that withholds decryption capability until time (T);

Verifier workflow via QR-encoded handles to DIDs/VCs for offline/online use;

Security and deployability discussion (threat model, failure modes, regulatory aspects).

The rest of the study is organized as follows: Section II reviews IEEE Biometric Open Protocol Standard (BOPS), Self-sovereign identity technology, the Horcrux protocol and the Elliptic Curves Time-Lapse Cryptography (ECTLC) protocol. Section III presents our new cryptographic protocol for decentralized biometric-based self-sovereign identity with time-lapse encryption called the ECTLC-Horcrux protocol and considers possible use cases. Section IV presents the results. Section V details the discussion. Finally, Section VI summarizes the study.

II. RELATED WORKS AND POSITIONING

1) SSI and verifiable credentials: Surveys and standards (including W3C Verifiable Credentials v2.0 and DID Core) systematize identifiers, credential models, and wallet architectures, highlighting selective disclosure and revocation

challenges. We adopt this stack to enable issuer-agnostic verification and offline validity checks [28, 29, 30].

2) Decentralized biometrics: Horcrux proposes cryptographic splitting of biometric templates to remove single points of failure. Our design reuses the splitting intuition but couples it with SSI wallets and time-based access control.

3) Time-lapse / Time-lock cryptography: ECC-based time-lapse cryptography enforces decryption only after (T) by threshold key release. We instantiate an ECC-based service and align its trust assumptions with SSI verifiers.

4) Biometric protection standards: ISO/IEC 24745:2022 provides guidance on biometric information protection; we comply by avoiding long-term template custody at the verifier and enforcing secure transport [31].

5) Positioning: Unlike prior work treating these threads separately, ECTLC-Horcrux composes SSI, decentralized biometrics, and ECC-based time-lapse into a single protocol with QR-mediated verifier flows and explicit threat considerations.

III. Methods

The protocol designed in this work is based on 2410-2017 - IEEE Biometrics Open Protocol Standard (BOPS), Self-Sovereign Identity technology, the Horcrux protocol, the Elliptic Curves Time-Lapse Cryptography (ECTLC) protocol, and the use of QR codes.

A. Biometrics Open Protocol Standard (BOPS) [6]

2410-2017 - IEEE Biometrics Open Protocol Standard (BOPS) is an open protocol designed to provide secure and trusted biometric authentication that meets the high requirements of national and international standards. The protocol covers the full cycle of working with biometric data - from their collection and storage to transmission and comparison, while implementing a multi-level security architecture.

At the Collection stage, BOPS defines the use of standard APIs for obtaining biometric templates (e.g., fingerprints, face, voice) using hardware trusted environments such as HSM (Hardware Security Module), TEE (Trusted Execution Environment) or TPM (Trusted Platform Module). The use of these modules ensures that data is collected and stored in isolated, inaccessible or encrypted memory, which significantly reduces the risk of leakage or unauthorized access.

For storage (Storage) of templates, BOPS provides for the use of secure formats and cryptographic containers with hardware encryption support. Moreover, the standard supports the cryptographic sharding mechanism, in which the biometric template is divided into several parts. One of them is stored locally on the device, and the second is either also stored locally or transmitted to a remote platform. In this case, the compromise of one of the parts does not allow the restoration of the original biometric template, which ensures resistance to leaks.

Transmission in BOPS is implemented via the Representational State Transfer (REST) interface protocol, and no biometric template can be transmitted in unencrypted form. Before sending, the data is wrapped in a cryptographic container encrypted using the server's public key generated at the registration stage. Transmission is carried out exclusively through a two-way secure TLS channel, and the client TLS certificate is installed on the mobile device in advance, at the time of installing the application.

At the Processing stage, the protocol prohibits long-term storage of templates. All matching operations are performed exclusively in RAM or in a local HSM, without using file systems, databases or other forms of persistent storage. This significantly reduces the risk of data compromise in the event of a system breach.

The BOPS protocol includes two key phases of operation: registration and authentication. In the registration phase, the server generates a key pair (RKP) by transmitting a public key to the device. On the client side, a local key pair (LKP) is generated, and the collected biometric template (initial biometric vector - IBV) is associated with this pair. The template and keys are then encrypted using RKP and transmitted to the server over a secure channel.

In the authentication phase, the protocol provides for three configurations:

1) Local match: a candidate biometric template (CBV) is collected on the device, which is matched with a reconstructed IBV collected from local and, if necessary, remote shares. The result of the match (a Boolean or numeric threshold) is encrypted and sent to the server. This configuration complies with FIDO UAF requirements, provided that a certified local authenticator is used.

2) Remote match: The CBV is encrypted and sent to the server, where it is matched against the corresponding portions of the IBV available to the server and sent from the client.

3) Combined authentication: The server can, upon request, encrypt its portions of the IBV and send them back to the client, where the match occurs. Once the match is complete, all biometric data and keys are destroyed from RAM.

To protect against man-in-the-middle (MITM) and replay attacks, BOPS uses one-time passwords and server challenges wrapped in encrypted crypto containers (envelopes), ensuring the integrity and authenticity of the transmitted data. Compared to other protocols such as FIDO UAF, BOPS offers comparable levels of security but has additional capabilities, such as remote storage, flexible sharding, and multi-level authentication, making it a more versatile solution for a variety of scenarios, including mobile platforms, cloud services, and IoT devices. However, the choice of storage and mapping mode may depend on regulatory constraints, institutional policies, and trust level requirements.

B. Self-Sovereign Identity [1-3]

Self-Sovereign Identity (SSI) is a conceptual and technical approach that gives users full control over their digital identity

without dependence on centralized certifying authorities. Traditional identity models based on third-party identity providers (such as governments, banks, and commercial platforms) come with risks of compromise, centralized control, and privacy breaches. SSI addresses these shortcomings by transferring control over identity directly to the user, while ensuring verifiability and trust based on cryptographic principles. SSI is based on a model in which identifiers and their verifiable credentials are stored by the identity subject (the user) and can be provided on request to various verifiers.

The key components of the technology are the following elements:

Decentralized identifiers (DIDs) are unique, persistent identifiers that do not depend on central registries. They are managed by the user and allow the creation of cryptographically verifiable links between entities [29]. A DID is associated with a DID document containing public keys, authentication methods, and service endpoints that allow other entities to interact with the owner of the identifier.

Verifiable Credentials (VC) are digital assertions signed by an issuer (e.g., an educational institution, government agency, or employer) that a user can present to a verifier [28, 30]. Verification is performed without a request to the issuer, solely by checking the digital signature and metadata of the credential.

A digital identity wallet is a software solution (mobile or desktop) designed to securely store DIDs, private keys, and received credentials. The wallet provides an interface for interaction with issuers and verifiers, implements the functions of consent to data transfer and supports exchange protocols.

To ensure confidentiality, authenticity, and other security properties, cryptographic methods are used, including asymmetric encryption and digital signatures based on public key cryptography to ensure the authenticity of credentials; zero-knowledge proofs to confirm certain attributes (e.g. age, affiliation with an organization) without disclosing additional data.

Decentralized ledgers (ledger-based DID methods) are used to publish and resolve DIDs. Depending on the architecture, implementations based on blockchain platforms (e.g. Hyperledger Indy, Ethereum, Sovrin) or lighter methods such as DID:Web are possible. Cost efficiency is achieved through Layer-2 solutions, notably the Polygon Amoy testnet for DID anchoring. This reduces on-chain operations to ~0.002 USD per transaction while maintaining SSI revocation capabilities [27]. The implementation of the technology is provided by the W3C consortium standards, including DID Core and Verifiable Credentials Data Model.

The SSI architecture is built on a three-party interaction model, including the following roles: Issuer (an organization that creates and signs digital certificates), Holder (an entity that manages its digital identity, stores credentials in a wallet, and provides them upon request), Verifier (a party that requests and verifies the authenticity of the information received). The interaction between the parties is implemented through the exchange of verifiable credentials (Verifiable Credentials) and decentralized identifiers (DIDs).

1) The Issuer creates and signs a certificate (VC) containing a statement (e.g., "The user graduated from University N"), as well as its DID.

2) The User (Holder) receives this certificate and stores it in its digital wallet associated with its own DID and keys.

3) If necessary, the User presents the certificate to the Verifier, providing proof (possibly with zero knowledge), after which the Verifier verifies the VC signature, receives the Issuer's public key through the DID ledger, and decides on the authenticity of the presented certificate.

This three-party protocol is built without constant dependence between the participants, which allows for offline validation and increased resistance to censorship or denial of service.

C. The Horcrux Protocol [4]

The Horcrux protocol is a secure and decentralized biometric identification method within the Self-Sovereign Identity (SSI) concept. It combines the advantages of biometric methods, the 2410-2017 - IEEE Biometric Open Protocol Standard (BOPS), and the decentralized identifier (DID) infrastructure. The main idea of the protocol is to separate and decentralize the storage of user biometric data with subsequent authentication through their joint processing. This eliminates the need for centralized storage, reducing the risk of leakage and providing the user with full control over their biometric templates. The Horcrux protocol is based on dividing the user's biometric template into two encrypted parts (shares) using visual cryptography techniques. One part is stored locally on the user's device, and the other is stored in a decentralized storage associated with the DID document. This ensures the absence of a single point of failure and increases resistance to data compromise.

The protocol consists of two parts: enrollment and authentication. The enrollment process includes the following components. First, there is the collection of biometric data. The user provides biometric data (e.g. fingerprint or face image), which is converted into an initial biometric vector (IBV). Next comes splitting and encryption. The IBV is split into two shares, each of which is encrypted. One part is stored on the user's device, the other in a DID document placed in a decentralized storage (e.g. IPFS, Dropbox, Google Drive). Then a DID is created. A unique decentralized identifier (DID) is generated, which refers to the corresponding DID document containing the encrypted part of the biometric data. Finally, there is the registration in the blockchain. The DID and its corresponding DID document are registered in the blockchain, ensuring the immutability and verifiability of the data.

The authentication process consists of the following components. First, there is the collection of current biometric data. The user provides biometric data, which is converted into a candidate biometric vector (CBV). Shares extraction and merging are then performed. The local share of the IBV is combined with the share extracted from the DID document,

and then compared with the CBV to verify the identity. Finally, authentication is performed. If the combined IBV matches the CBV, authentication is considered successful.

D. The Elliptic Curves Time-Lapse Cryptography (ECTLC) Protocol [5]

The Elliptic Curves Time-Lapse Cryptography (ECTLC) protocol is based on the Time-Lapse Cryptography (TLC) protocol [12]-[14], the protocol for encrypting client data in such a way as to ensure that decryption cannot occur before a certain designated time, regardless of whether the sender wishes it. TLC protocol includes Pedersen's distributed key generation protocol [15], Feldman's threshold verifiable secret sharing protocol [16], and ElGamal encryption [17]. The TLC protocol uses the agreed-upon parameters of the ElGamal encryption algorithm, including the prime number p, the generator g of prime order q. These parameters can be found, for example, in RFC 3526 [18] and RFC 5114 [19]. The ECTLC protocol uses similar elliptic curve cryptography algorithms, namely the distributed key generation protocol based on discrete logarithm on elliptic curves [20], the Pedersen verifiable threshold secret sharing protocol [21], and the ElGamal elliptic curve encryption algorithm [22], respectively. The protocol is also implemented using a Service (Time-Lapse Cryptography Service) consisting of n participants P_1, ..., P_n. Each participant of the Service P_i can be represented as an autonomous computer (server) that accurately and secretly performs the calculations required by the protocol, securely stores all its secret data, and has a secure way of backing up the data for emergency recovery. All participants of the Service can privately and secretly exchange information with each other, forming a network. A threshold value t is assumed such that at most $t - 1$ participants can violate the protocol, and at least t participants are reliable. The condition $n \geq 2t - 1$ ($t \leq (n+1)/2$) must be satisfied, e.g., if $n = 3$, then $t \leq 2$.) For further efficiency, reliability, and resistance to attacks, a small network M of K managers is used, which acts as a "control team" for the Service. This control team must create a schedule of public and corresponding private keys generated by the Service; maintain an internal bulletin board for use by Service participants; maintain an open bulletin board for Service users. The integrity of these bulletin boards is ensured by each manager maintaining their own copies of the two bulletin boards. Service participants and users will look at messages posted on each copy of the bulletin boards and determine the correct values by the majority of entries. Each Service participant digitally signs each message. The Service may generate key structures on a periodic basis; for example, it may create keys with a lifetime of 1 week every day, or create keys with a lifetime of 2 hours every 30 minutes. Such a schedule is posted by managers on an open bulletin board. In addition, the Service may accept requests from users to generate new keys with a specified lifetime; managers accept these requests and post them on an open bulletin board. Service participants create keys according to the protocol, sign them, and publish the signed key structures on an open bulletin board. The actions of all protocol participants are synchronized using publicly available and reliable clocks such as those provided by NIST. The protocol provides for the use of agreed parameters of the elliptic curve used: the modulus of the elliptic curve, a prime number p, the equation of the elliptic curve, the coefficients of the equation a and b from the field Fp, and a point of the elliptic curve G of prime order q. These parameters can be found, for example, on the SafeCurves project website [23].

E. QR Codes [24-25]

Two-dimensional matrix barcode technology, QR (Quick Response), is a means of presenting, transmitting and verifying structured digital data. Due to their compactness, high encoding density, resistance to damage and widespread support in software and hardware, QR codes are widely used in modern information systems to ensure contactless and offline transactional transfer of information. A QR code is a two-dimensional image consisting of black and white modules (squares) located in a square grid. It encodes binary or symbolic information that can be read by an optical scanner (including mobile device cameras) and converted back into the original data. Modern QR codes allow you to encode up to several thousand characters in a single image, including text, JSON structures, URLs, digital signatures and other formats. QR codes can be used to transmit structured data (e.g. JSON representation of digital certificates, access tokens, session keys, etc.), user identification (generation of personalized codes containing DID or hash identifiers for authentication), offline interaction (enabling the exchange of information between entities without a network connection), data signing and verification (inclusion of digital signatures in encoded data to ensure its authenticity when scanned), generation of passes, tickets, vouchers and transaction messages (visualization of one-time or permanent digital passes with the possibility of machine verification).

F. Biometric Data Protection Via Steganography

To ensure secure transmission of SSI credentials within TLC frameworks, QR-encoded Verifiable Credentials (VCs) and Decentralized Identifiers (DIDs) are embedded into facial images using LSB steganography. Each credential layer is distributed across a dedicated RGB channel (e.g., VC in R, DID in G), enabling standard QR decoders to extract payloads while maintaining visual fidelity (distortion <0.1%, as validated in prior research [26]). Steganography is used only as a transport/masking channel for payloads and does not affect the cryptographic access policy; it is orthogonal to TLC/SSI and therefore optional

IV. Results

The ECTLC-Horcrux protocol describes the interaction of the following participants: the User (Holder), the ECTLC-Service, the Service Provider (Issuer), the Service Provider (Verifier), the BOPS Server A, the BOPS Server B, the Blockchain, and the Identity hub. The protocol consists of two stages: enrollment and authentication.

Step-by-step description of the protocol.

A. Enrollment

1) The User produces his biometric template (initial biometric vector) using his mobile device.

2) The produced initial biometric vector is encrypted into two shares via visual cryptography by the mobile device.

3) One share is secured on the mobile device; the second share is encrypted with the device's private key.

4) The encrypted share, the corresponding device's public key and a time value T before which access to biometric data cannot be granted are sent to the Issuer.

5) The Issuer requests the ECTLC-Service for the ECTLC-Service's public key, indicating the time value T.

6) Upon receiving the request, the ECTLC-Service generates a public key according to the ECTLC cryptographic protocol and transmits it to the Issuer.

7) The Issuer encrypts the received encrypted share with the ECTLC-Service's public key and sends it with the device's public key to the BOPS Server A.

8) The BOPS Server A creates a DID and the DID document, and stores the DID on the Blockchain, and stores the DID document on the Identity hub.

9) The BOPS Server A sends the created DID to the Issuer.

*10)*The Issuer sends the received DID to the mobile device.

B. Authentication

1) The User creates a QR code with his DID, signature and public key into the Verifier using his mobile device and sends it to the Verifier.

2) The Verifier scans the received QR code. Should initial decoding fail (e.g., due to partial image corruption), the system initiates recursive grid-based scanning to reconstruct the SSI payload from salvageable segments. Upon successful recovery, it requests the corresponding DID document from the BOPS Server B.

3) The BOPS Server B resolves the DID and fetches the corresponding DID Document via a Blockchain from the Identity hub.

4) The BOPS Server B requests the ECTLC-Service for the ECTLC-Service's private key.

5) Upon receiving the request, the ECTLC-Service checks the time value. If the current time value has already reached the time value T, the ECTLC-Service generates a private key according to the ECTLC cryptographic protocol and sends it to the BOPS Server B. If the current time value has not yet reached the time value T, the ECTLC-Service rejects the request and sends the BOPS Server B a message that data is not available until the specified time value. And the BOPS Server B sends the corresponding message to the Verifier.

6) The BOPS Server B verifies known issuer signature and enrollment signature, and decrypts encrypted share with the received ECTLC-Service's private key.

7) The BOPS Server B requests the secured share and signed challenge from the mobile device.

8) The secured share is retrieved.

9) The User produces his biometric query template using his mobile device.

*10)*The challenge is signed with the device's private key.

*11)*The retrieved first share, query template and signed challenge are sent to the BOPS Server B.

*12)*The BOPS Server B verifies the challenge signature and decrypts the encrypted share using enrollment public key.

*13)*The BOPS Server B combines shares and performs match.

*14)*The BOPS Server B returns the match result to the Verifier and deletes shares and associated keys.

*15)*The Verifier authorizes access.

The proposed protocol enables a wide range of scenarios where time-sensitive access to identity documents and personal data, including biometrics, is required. The following are some possible use cases where delayed access is critical.

Example 1. A will with biometric confirmation after death.

The user creates a digital will encrypted using TLC and stored in a decentralized repository. The owner's biometric template (e.g., fingerprint) is shared using the Horcrux scheme. Access to the decrypted will is possible only after legal confirmation of death and the expiration of a set time interval (e.g., 30 days). This model prevents premature disclosure of the will and ensures that only authorized persons can access it.

Example 2. Medical data with access at the time of surgery.

The patient uploads preliminary consent for surgery and their medical data in encrypted form. The documents are only available at the time of surgery and are automatically activated at the appointed time via TLC. Access is only possible after the patient and medical staff have undergone biometric verification. This protects against manipulation and premature familiarization of medical information by third parties.

Example 3. A court opinion activated at a hearing.

A digital expert opinion (e.g., on biological evidence) is encrypted by TLC and published in the blockchain with an activation tag on the court date. A QR code confirming the availability and authenticity of the opinion is presented for verification in court. The expert's biometrics additionally confirm authorship, and the impossibility of disclosure before the specified deadline excludes any influence on the process.

Example 4. A digital pass with future activation.

As part of physical access to a facility (e.g., a data center), the contractor receives a VC with the right to enter, but it is activated only from a certain fixed time. The contractor's biometrics are used as a means of identity verification, and the QR code contains a link to encrypted access rights. Until activation, access is impossible, either technically or cryptographically.

V. DISCUSSION

1) Threat model and mitigations: We consider MITM and replay; the verifier requires signed envelopes and challenge–response over authenticated channels. A single biometric share compromise does not enable reconstruction; premature disclosure is prevented by the ECTLC time barrier.

2) Limitations: Threshold availability of ECTLC participants and DID/VC revocation signaling are operational dependencies; biometric capture quality may impact FAR/FRR.

3) Applicability: Medical/legal attestations and controlled physical access where provable non-access before time (T) is a strict requirement.

VI. Conclusion

The study proposes a cryptographic protocol that expands the capabilities of Self-Sovereign Identity by supporting deferred access to data and decentralized management of biometric information. Based on a combination of the Horcrux and the Elliptic Curves Time-Lapse Cryptography protocols, the developed approach ensures a high level of privacy and security, and complete user autonomy in managing their identities. The protocol is applicable in legal, medical, educational and commercial scenarios where strict time control and provability of digital identity are required. The proposed protocol takes into account current threats in the field of digital identification and privacy of personal data. Due to the use of Time-Lapse Cryptography, data protected within the Verifiable Credential is inaccessible until the moment T specified during encryption. Even with an encrypted message, an attacker is unable to access the content until the cryptographic "delay" is complete. Instead of storing templates in centralized databases, the Horcrux protocol is used, which provides distributed storage and restoration of biometrics only by multilateral agreement. This eliminates the possibility of compromising templates even if one of the participants is hacked. The use of VC and DID signed via blockchain allows for verification of authorship and immutability of data. QR codes contain only links to data and digital fingerprints, eliminating the possibility of falsification without access to the Issuer's private key. In the future, it is planned to implement a reference implementation and detailed analysis of the protocol for performance and correctness.

Acknowledgment

The work was carried out with the financial support of the Science Committee of the Ministry of Science and Higher Education of the Republic of Kazakhstan, No. AP19678000.

References

[1] Preukschat, A., & Reed, D. (2021). Self-Sovereign Identity: Decentralized digital identity and verifiable credentials. Manning Publications.

[2] Ferdous, M. S., Chowdhury, F., & Alassafi, M. O. (2019). In search of self-sovereign identity leveraging blockchain technology. IEEE Access, 7, 103059–103079. https://doi.org/10.1109/ACCESS.2019.2931173.

[3] Mühle, A., Grüner, A., Gayvoronskaya, T., & Meinel, C. (2018). A survey on essential components of a self-sovereign identity. Computer Science Review, 30, 80–86. https://doi.org/10.1016/j.cosrev.2018.10.002.

[4] Harmon, J. D., Tobin, A., & Reed, D. (2019). The Horcrux Protocol: A Method for Decentralized Biometric-based Self-sovereign Identity. In Proceedings of the 2019 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPCON) (pp. 37–46). IEEE. https://doi.org/10.1109/DAPPCON.2019.00013.

[5] Tasmagambetov, Olzhas, Yerzhan Seitkulov, Ruslan Ospanov, and Banu Yergaliyeva. “Fault-tolerant Backup Storage System for Confidential Data in Distributed Servers.” TELKOMNIKA (Telecommunication Computing Electronics and Control) 21, no. 5 (October, 2023): 1030. https://doi.org/10.12928/telkomnika.v21i5.25305.

[6] 2410-2017 - IEEE Standard for Biometric Open Protocol Standard. IEEE. https://doi.org/10.1109/IEEESTD.2017.8089818.

[7] Cui, H., Whitty, M., Miyaji, A., Li, Z. (2025). A Blockchain-Based Digital Identity Management System via Decentralized Anonymous Credentials. In Proceedings of the 6th ACM International Symposium on Blockchain and Secure Critical Infrastructure (BSCI '24). Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/3659463.3660027.

[8] Paredes-García, D., Fernández-Carrasco, J. Á., López, J. A. M., Vasquez-Correa, J. C., Yoldi, I. J., Moreno-Acevedo, S. A., González-Docasal, A., Irazusta, H. A., Muniain, A. Á., & Loinaz, Y. d. D. (2025). SIBERIA: A Self-Sovereign Identity and Multi-Factor Authentication Framework for Industrial Access. Applied Sciences, 15(15), 8589. https://doi.org/10.3390/app15158589.

[9] Saprunov, V., Muhammad, F., Kyung-Hyune, R. (2024) Privacy-Preserving Decentralized Biometric Identity Verification in Car-Sharing System. J Multimed Inf Syst 2024;11(1):17-34. https://doi.org/10.33851/JMIS.2024.11.1.17.

[10] Naicker, D., Moodley, M. (2024) Challenges of user data privacy in self-sovereign identity verifiable credentials for autonomous building access during the COVID-19 pandemic. Front. Blockchain 7:1374655. https://doi.org/10.3389/fbloc.2024.1374655.

[11] Enge, A., Satybaldy, A, Nowostawski, M. (2022) An offline mobile access control system based on self-sovereign identity standards. Computer Networks, Volume 219, 109434, https://doi.org/10.1016/j.comnet.2022.109434.

[12] Rabin, M.O., Thorpe, C.A. (2006) “Time-lapse cryptography”, Technical report TR-22-06, Harvard University School of Engineering and Computer Science.

[13] Rabin, M.O., Thorpe, C.A. (2007) “Method and apparatus for time-lapse cryptography”, U.S. Patent 8,526,621.

[14] Thorpe, C.A., Barrientos, M., Rabin, M.O. (2009) “Implementation of A Time-Lapse Cryptography Service”, IEEE Symposium on Security and Privacy, Oakland.

[15] T. P. Pedersen. A Threshold Cryptosystem without a Trusted Party. In Advances in Cryptology—Eurocrypt’91, pages 522–526. Springer-Verlag, 1991.

[16] P. Feldman. A practical scheme for non-interactive verifiable secret sharing. IEEE Symposium on Foundations of Computer Science, pages 427–437, 1987.

[17] T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans. Information Theory, IT-31(4):469–472, 1985.

[18] Kivinen, T. M. Kojo, "More Modular Exponential (MODP) Diffie-Hellman groups for Internet Key Exchange (IKE)", RFC 3526, May 2003, doi: https://doi.org/10.17487/RFC3526

[19] M. Lepinski, S. Kent “Additional Diffie-Hellman Groups for Use with IETF Standards”, RFC 5114, January 2008, doi: https://doi.org/10.17487/RFC5114

[20] Tang C., Chronopoulos, A.T. An Efficient Distributed Key Generation Protocol for Secure Communications with Causal Ordering // Proceedings of IEEE ICPADS 2005, The 11th International Conference on Parallel and Distributed Systems, 20-22 July 2005, Volume 2, Fukuoka, Japan, pp. 285 - 289.

[21] Pedersen T.P. Non-iteractive and information-theoretic secure verifiable secret sharing // In Proc. Of CRYPTO 1991, the 11th Ann. Intl. Cryptology Conf., vol. 576 of Lecture Notes in Computer Science, pp. 129–140. Aug. 1991.

[22] Trung M.M., Do L.P., Tuan D.T., Tanh N.V., Tri N.Q. Design a cryptosystem using elliptic curves cryptography and Vigenère symmetry key // IJECE, vol. 13, no. 2, pp. 1734-1743, Apr. 2023, doi: 10.11591/ijece.v13i2.pp1734-1743.

[23] Bernstein D.J., Lange T. SafeCurves: choosing safe curves for elliptic-curve cryptography. http://safecurves.cr.yp.to

[24] Kukharev, G., Kaziyeva, N., Tsymbal, D. (2018) Barcoding Technologies for the Tasks of the Facial Biometrics: State of the Art and New Solutions. Pattern Recognition and Image Analysis, vol. 28, pp. 496-509.

[25] Soldek J. et al. (1997) Image analysis and pattern recognition in biometric technologies. In: Proc. Int. Conf. on the Biometrics: Fraud Prevention, Enhanced Service. Las Vegas, USA, pp. 270-286.

[26] Kukharev G.A., Kaziyeva N. (2019) Algorithms of color QR codes formation for biometry tasks. Scientific and Technical Journal of Information Technologies, Mechanics and Optics, 2019, vol. 19, no. 5, pp. 955-958 (in Russian). doi: 10.17586/2226-1494-2019-19-5-955-958.

[27] Polygon Team, "Polygon Amoy Testnet Documentation," 2023. [Online]. Available: https://docs.polygon.technology.

[28] W3C. Verifiable Credentials Data Model v2.0. W3C Recommendation, 15 May 2025.

[29] W3C. Decentralized Identifiers (DID) v1.0. W3C Recommendation, 19 July 2022.

[30] W3C. Data Integrity BBS Cryptosuites v1.0. Candidate Recommendation Draft, 3 April 2025.

[31] ISO/IEC 24745:2022. Information technology-Security techniques - Biometric information protection.

Integrating Chatbots into E-Learning Platforms: A Systematic Review

Victor Sevillano-Vega, Juan Chavez-Perez, Carmen Torres-Ceclén, Orlando Iparraguirre-Villanueva*
Facultad de Ingeniería, Universidad Tecnológica del Perú, Chimbote, Perú

***Abstract*—The application of chatbots in e-learning has experienced rapid growth in recent years, but a dilemma remains about their pedagogical contribution in practice. For this reason, the aim of this systematic literature review was to analyze the implementation of chatbots in e-learning platforms, evaluating their benefits, academic impact and challenges. The methodology used was PRISMA 2020 (Preferred Reporting Items for Systematic Reviews and Meta-Analyses), based on a structured search in databases such as Scopus, Web of Science, Springer and ScienceDirect. The selection included 55 studies published between 2020 and 2024, after applying rigorous inclusion and exclusion controls. The research results show that personalization of learning, self-regulation, increased student engagement and educational efficiency benefit most when chatbots are integrated with active methodologies. Geographically, scientific output was dominated by the UK, Malaysia and Spain, with 38.18% of publications in 2024. It was also found that the majority of methodological approaches were quantitative, followed by mixed and qualitative studies less frequently. Among the barriers that emerged in terms of the pedagogical dimension were teacher resistance and limited training in artificial intelligence tools. Educational issues, privacy concerns, and biases in generated responses also emerged. Keywords from co-occurrence analysis using VOSviewer revealed the prominence of terms such as chatbot, intelligent tutoring and technology-enhanced learning in recent scientific output. Thus, it is concluded that chatbots are a determinant of autonomy, motivation and effectiveness of online learning, leading to a change in future educational environments, where students will adopt emerging technology. Among the limitations of this review were the scarcity of longitudinal studies and restricted access to certain articles.**

***Keywords*—Chatbots; educational platforms; e-learning; education; challenges**

I. Introduction

In recent years, the integration of chatbots into e-learning platforms has established its role in responding to the current educational challenges exacerbated by the COVID-19 pandemic. The United Nations Educational, Scientific and Cultural Organization (UNESCO) emphasizes that scientific and technological advances have enabled health crises to be mastered to a greater extent compared to historical pandemics, opening fertile ground for innovative educational solutions [1]. However, within this scenario of digital transformation, in which e-learning is growing rapidly with materials such as videoconferencing and interactive content [2], the adaptation of educational actors such as teachers, assistants and students to blended modalities continues to face significant obstacles [3].

The development of educational chatbots is considerable and shows enormous potential for transformation within various educational environments. A recent study [4] points out that common speech recognition application programming interfaces (APIs) have high error rates (WER/CER) in the processing of speech disorders, thus requiring custom adaptations for educational use. Also, research [5] based on the technology acceptance model (TAM) shows that students welcome 24/7 accessibility and the use of an intuitive interface by educational chatbots.

Applications integrated with advanced platforms are yielding significant results. Cloud computing [6] makes it possible to handle large volumes of queries simultaneously, while innovative solutions such as augmented reality chatbots [7] have a positive impact on the learning process: tests with 102 biology students reveal a substantial improvement in motivation according to the attention, relevance, confidence and satisfaction (ARCS) model, with an 80% preference over traditional methods [8]. Similarly, remarkable progress is observed within the educational environment, where a mobile application with augmented reality increased the proportion of interest by 100%, comprehension by 50% and achieved a satisfaction level of 40% at the level of Peruvian primary and secondary school students [9]. In contrast, technical analysis [10] reflects that natural language processing (NLP) models, subject to sequential neural networks, obtain higher levels of accuracy in chatbots for university use, surpassing approaches such as term frequency and inverse document frequency (TF-IDF) or pattern matching.

Practical implementation faces considerable obstacles. Research on mood using bi-directional encoder from transformer representations (BERT) models [11], with accuracy as high as 96.49%, raises serious concerns about the quality and potential bias of the generated feedback. Likewise, the accelerated growth of e-learning [12] and the progress of artificial intelligence (AI) [13] augur an increasing role for chatbots. In this sense, systems such as LearningPartnerBot [14], integrated in Moodle, employ the Felder-Silverman model in the personalization of content based on learning styles, facing the problem of cold start in recommendations.

This research is justified by the compelling need to synthesize the scientific literature related to educational chatbots, given their rapid post-pandemic growth and persistent challenges in technical accuracy and pedagogical integration. Furthermore, the goal of this SLR is to systematically analyze architectural features, pedagogical effectiveness, and implementation gaps to propose evidence-based guidelines to optimize the adoption of chatbots in e-learning.

II. Literature Review

A. Pedagogical Benefits of Educational Chatbots

The implementation of chatbots in education has proven to bring benefits in learning personalization. In research [15], an AI-based e-learning chatbot was designed to crawl multimedia content and keyword extraction techniques through NLP, achieving optimal indexing, decreased search time, and increased student satisfaction. Similarly, in [16], the authors analyzed a set of 2003 web articles, using web mining and machine learning (ML) methods to assess the existing public perception of ChatGPT in education, finding that it improves writing skills and enhances the dynamism of learning environments, although risks of plagiarism and ethical issues requiring regulation were identified. Complementarily, [17] examined the experience accumulated by 360 university students in Malaysia regarding the use of chatbots oriented to English learning, using a combined approach based on the Push-Pull Mooring Habit (PPMH) model; consequently, The results showed that simplicity of use and high performance were conducive to adoption, while social isolation concerns posed an obstacle.

B. Impact of Chatbots on Student Academic Performance

The impact of chatbots on academic performance is the subject of recent studies. The study [18] developed the MERLIN project to assist Malaysian university students in the online learning process, using an AI chatbot equipped with multimedia learning mechanisms; based on data collection from 102 students, they showed that the virtual assistant favored the ability to comprehend academic content. Similarly, in [19], they investigated the adoption of chatbots in higher education based on the diffusion of innovations theory, for which they surveyed 842 students, finding that relative advantages, compatibility, evidence of use, and trust exerted positive effects for intention to use, while perceived usefulness and operational ease did not show a direct relationship. Similarly, in the study of [20], the authors evaluated the use of ChatGPT applied to the learning process of full-time and part-time students using a comparative quantitative approach. They analyzed the degree of students' familiarity with the chatbot functionality and their ability to formulate queries. As a result, it was shown that full-time students took advantage of the linguistic model, while no significant differences in terms of experience with the functionality or query formulation capabilities were found between the two groups. As a result, it was shown that full-time students took advantage of the linguistic model, while no significant differences in terms of experience with the functionality or query formulation ability were found between the two groups of students.

C. Technical and Pedagogical Challenges in the Implementation of Chatbots

The technical and pedagogical difficulties involved in the implementation of chatbots in educational environments are the focus of several studies. In [21], exploratory research was carried out to determine the perceptions of 142 teachers of English as a second language (ESL) regarding the use of chatbots in teaching and mobile learning, using a sample design based on Likert-type surveys and open-ended questions. The results showed that teachers positively valued the use of chatbots to provide feedback and emulate interaction cycles; however, many expressed the need to receive additional training for their proper implementation. At the same time, [22] investigated the integration of ChatGPT together with interactive learning environments (ILEs) within the teaching of computer networks, carrying out a pilot experiment involving three graduate students, whose results showed a significant improvement in the level of student learning.

D. Ethical Considerations and Risks Associated with the Use of Chatbots in Education

The risks associated with the use of chatbots in academia and ethical issues highlight issues that need to be explored. In this scenario, [23] investigated the impact of knowledge management factors on the degree of satisfaction and continuity of ChatGPT use among university students in Vietnam, employing a quantitative methodology, showing a confirmation of expectations with a positive and significant effect on perceived usefulness and satisfaction, increasing the continuity of tool use. Complementarily in [24], they evaluated the applicability of ethical principles in the use of AI tools such as ChatGPT performed on students aged 12 to 24 years, by using a quantitative analysis based on the technology acceptance model, evaluated through a structural study the effect of ethical principles such as usefulness, fairness, privacy and data protection on perceptions about the use of ChatGPT, evidencing that younger students between 12 and 18 years of age of the general range of study, obtained responses from ChatGPT to support mainly their decisions, but also expressed concern about the use of their personal data.

TABLE I. Major findings and limitations in research on educational chatbots.

Ref.s	Main results	Limitations
[15]	NLP chatbot improved content search and student satisfaction.	Did not evaluate prolonged pedagogical impact.
[16]	Improved writing and interactive environments; generated ethical risks.	Based on perceptions without classroom validation.
[17]	Ease of use drove adoption; robotic interaction caused isolation.	Geographically limited sample.
[18]	Assistant improved academic comprehension during pandemic.	Small sample and exceptional context.
[19]	Relative advantages determined adoption; usefulness did not correlate with use.	Self-report without behavioral verification.
[20]	Full-time students took better advantage of language model.	Disciplinary variables not controlled.
[21]	Teachers valued feedback but needed additional training.	Uncorrelated with learning outcomes.
[22]	Improved comprehension, but presented problems in interfaces.	Minimum sample and insufficient duration.
[23]	Knowledge acquisition determined satisfaction and continued use.	Specific cultural context not generalizable.
[24]	Ethical curriculum improved understanding; adolescents distrusted data	Wide age range diluted differences.

Table I summarizes the main findings and limitations identified in the studies reviewed on the use of chatbots in educational contexts.

III. METHODOLOGY

This SLR comprehensively addressed how chatbots can drive the transformation of e-learning platforms, with an analysis of their pedagogical benefits, their impact on academic performance and the challenges presented by their implementation. The integrated methodological approach contemplated four essential components: 1) formulation of research questions based on the identified knowledge gaps, 2) application of the PRISMA protocol to ensure scientific rigor, 3) systematic search strategy in multiple databases, and 4) standardized selection criteria that guarantee the quality of the evidence analyzed.

A. Purpose and Research Questions

Preliminary research of scientific literature revealed three critical gaps in the field of educational chatbots: insufficient conclusive evidence of their pedagogical impact; lack of documentation on the difficulties of their application; and lack of contrasted studies on their effectiveness in different online learning scenarios. To address these shortcomings, the following research questions emerged:

- What are the benefits of integrating chatbots into online platforms?
- How does the integration of chatbots in e-learning platforms affect academic performance?
- What are the technical and pedagogical challenges faced by educational institutions when implementing chatbots in their e-learning platforms?

B. Type of Study

For the methodological development of this research, the approach based on systematic literature reviews was adopted, as established by the PRISMA 2020 protocol, an international norm recognized as a reference standard in terms of transparency and scientific rigor for this type of review [25]. By applying this protocol, it is possible to establish a precise and replicable operating structure, reinforcing the validity of the results through well-defined procedures in the phases of identification, screening, selection and synthesis of the studies. In addition, as a visual documentation tool, the PRISMA flowchart was implemented to transparently detail the entire bibliographic selection process. The use of this resource allowed the traceability of the different decisions taken during the filtering of the documents, transparently showing the procedure for applying the inclusion and exclusion criteria at each moment [26].

C. Search Strategy

The evidence collection work was carried out using a systematic search strategy, phased according to the recommendations of the PRISMA 2020 protocol. For this purpose, four academic databases of recognized prestige were chosen, such as Scopus, Springer, ScienceDirect and Web of Science, whose coverage of publications on educational technology and intelligent learning systems was notable. The search sequence was designed by combining Boolean operators and terms of reference derived from the research questions. The syntactic format used was the following: ("chatbot" OR "conversational agent" OR "dialog system") AND ("e-learning" OR "online education" OR "digital learning") AND ("LMS" OR "learning management system" OR "educational platform") AND ("challenge" OR "limitation" OR "implementation barrier").

In accordance with the PRISMA 2020 protocol, the evidence collection work was structured in four main phases:

- Identification: Exhaustive search in selected databases to identify relevant studies.
- Selection: Elimination of duplicates and preliminary review of titles and abstracts.
- Eligibility: Full-text evaluation to ensure relevance to the research questions.
- Inclusion: Final selection of studies that met all the established inclusion criteria.
- As a direct result of this process, 563 studies were initially collected, and 55 studies were identified and finally selected that fully met the criteria defined for the review and constituted solid evidence to respond to the objectives set. Fig. 1 shows the sequential distribution of the entire procedure under the PRISMA methodology.

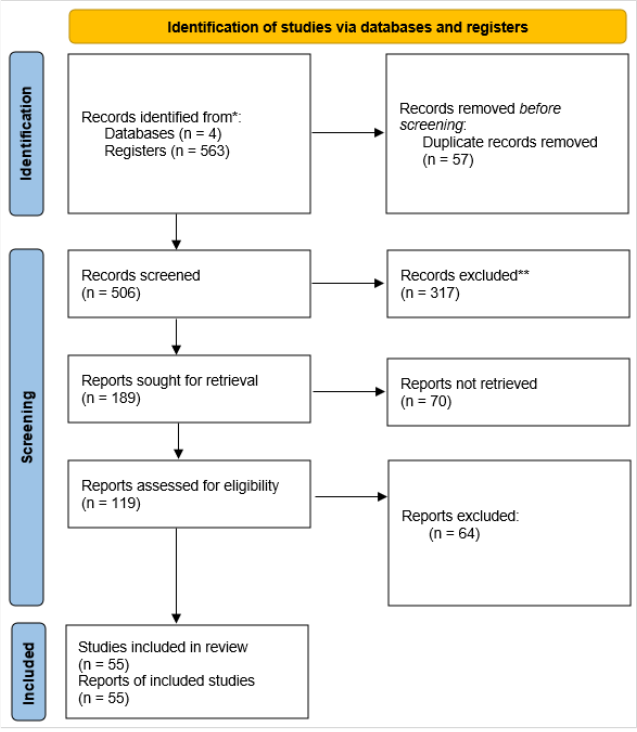


Fig. 1. PRISMA methodology.

D. Inclusion and Exclusion Criteria

For the development of this systematic review, inclusion and exclusion criteria were previously established to ensure the coherence, relevance and methodological soundness of the

corpus analyzed. The criteria allowed literature to be filtered objectively, directing the selection to empirical studies strictly related to the application of chatbots in e-learning environments. In this way, it was possible to minimize biases, guarantee the reproducibility of the process and reinforce the validity of the results. The criteria considered are presented below:

1) Inclusion

- Studies on the use of chatbots in e-learning platforms.
- Empirical research with educational results in real contexts.
- Publications between 2020 and 2024.
- Documents in English and with access to the full text.

2) Exclusion

- Studies outside the field of education or e-learning.
- Non-original and non-empirical papers such as reviews, editorials, theses and chapters.
- Non-English language publications.
- Duplicate or non-peer-reviewed records.
- Research that does not address benefits, academic impact or challenges of using chatbots in e-learning.

IV. Results

The study selection process, carried out under the PRISMA 2020 guidelines, began with the identification of 563 records in the Scopus, Web of Science, ScienceDirect and Springer databases. After applying inclusion/exclusion criteria and thematic filters, the corpus was refined to 55 studies published between 2020 and 2024, considered relevant to analyze the integration of chatbots in e-learning platforms. Fig. 2 illustrates the complete flow of the initial collection in the research, as well as the result obtained in the final selection.

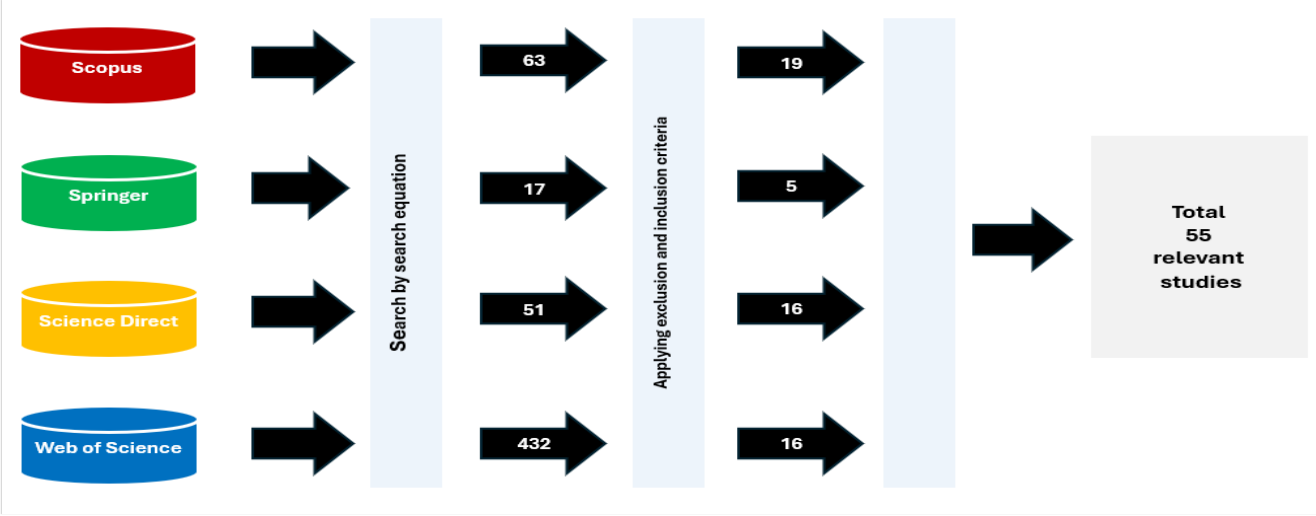


Fig. 2. Distribution of studies collected and final selection.

In the preliminary phase, 57 duplicates were eliminated, leaving 506 studies for the initial analysis. The first screening phase consisted of an exhaustive analysis of titles, where 317 articles that did not meet the thematic criteria were excluded, particularly those that did not mention educational applications of chatbots. As a result of this process, a new set of 189 studies was obtained, showing the following distribution: Scopus with 42 studies in 22.22%, Springer with 9 in 4.76%, ScienceDirect with 35 in 18.52% and Web of Science showing the predominance with 103 studies with a value of 54.50%, highlighting as the main source of academic information on the subject.

In the second phase, the analysis of abstracts allowed the exclusion of 70 additional articles that did not meet the requirements of thematic alignment with the research objectives, specifically those that did not address chatbot e-learning integration, lacked demonstrable pedagogical components or presented limitations of access to the full text. This rigorous filtering process reduced the corpus to 119 potentially relevant studies.

The final evaluation focused on three key aspects derived from the research questions: the educational benefits of chatbots, their measurable impact on academic performance, and the technical and pedagogical challenges identified in their implementation. After a thorough analysis of the full texts, 64 studies that did not provide direct evidence on at least one of these key aspects were eliminated, resulting in a final selection of 55 studies that strictly met the established criteria.

The final selection of 55 studies revealed that Scopus contributed the most with 19 studies (34.55%), followed by ScienceDirect with 16 studies (29.09%), Web of Science with 15 studies (27.27%), and Springer with 5 studies (9.09%). Fig. 3 provides the proportional representation of each database in a bar chart, presenting the result for the respective visual analysis.

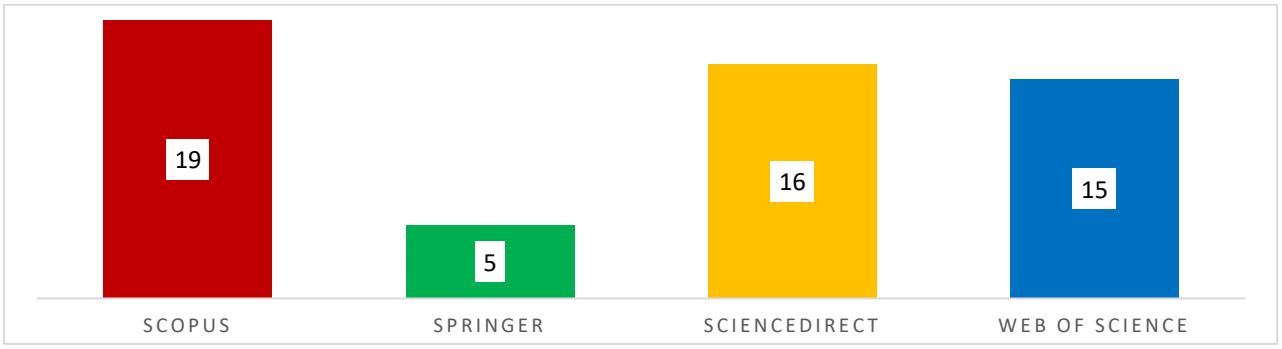


Fig. 3. Distribution of studies by database.

To analyze the articles by year range, it was determined that each publication complies with the period from 2020 to 2024. The following percentage graph shows the 55 articles corresponding to each year: 38.18% in 2024, 29.09% in 2023, 20% in 2022, 5.45% in 2021 and 7.27% in 2020. As shown in Fig. 4, the year 2024 shows a peak in the number of publications on the subject, followed by 2023, with 2021 having the lowest peak among this range of years.

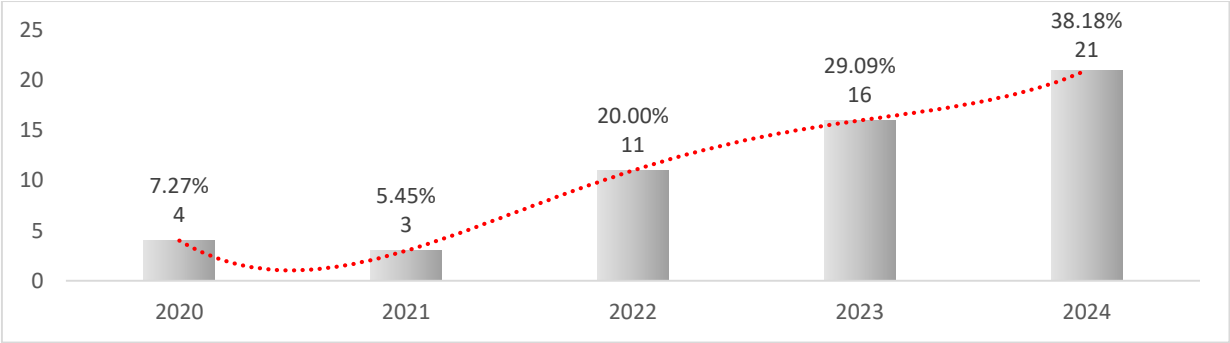


Fig. 4. Distribution of studies by year of publication.

On the other hand, Fig. 5 identifies those studies that correspond to their corresponding years. In 2020, 5.26% of the studies came from Scopus and 20% from Web of Science. For 2021, Scopus presented a slight decrease to 5.26%, while Web of Science obtained 13.33%. In 2022, a significant increase was observed in Scopus, reaching 42.11%, while Web of Science maintained 20%. In 2023, Scopus decreased by 26.32%, while Springer reached 40% and ScienceDirect with 37.50%, being the most representative in this year, while Web of Science maintained its 20%. Finally, in 2024, ScienceDirect increased to 62.50%, Springer reached 60%, while Scopus decreased by 21.05% and Web of Science increased by 26.67%. This figure shows the distribution of studies by each database and year.

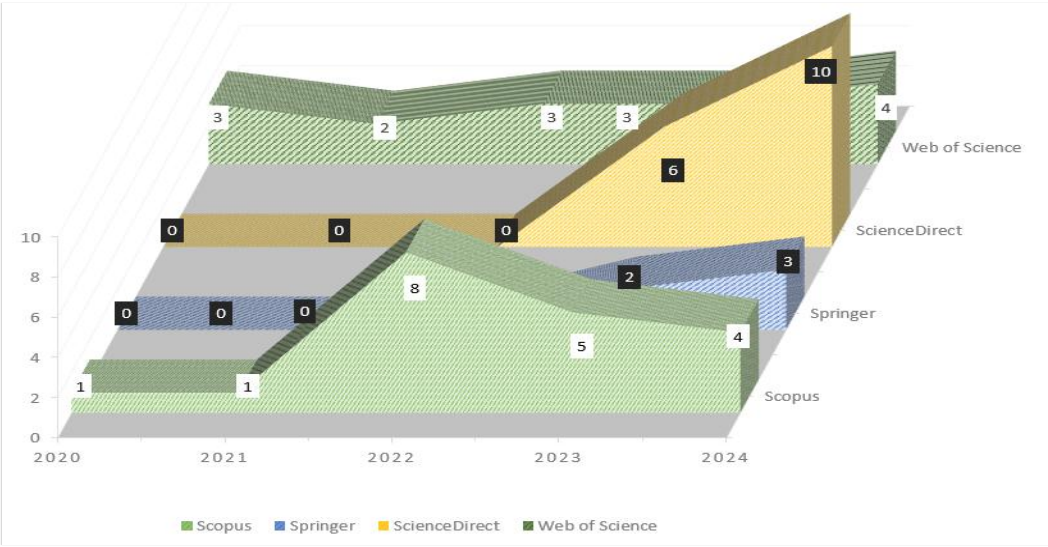


Fig. 5. Number of studies by year and databases.

In the geographical distribution of studies on the subject, the analysis identified the countries that stand out the most in terms of educational chatbots, the United Kingdom with 10.91%, followed by Spain, Taiwan and Malaysia with 7.27%, respectively. Meanwhile, the countries of Canada, Germany, Saudi Arabia, Africa, Turkey and the United States have 3.64%. As shown, there is an overview of the articles that stand out most among these countries and a predominance in the English language, whose inclusion criterion was defined for this work, which makes this information more accessible. These results are expressed in Fig. 6 below.



Fig. 6. Reviews of the scientific literature by country.

Malaysia and Spain are important because of their policies that support educational innovation, their financial resources, and their strong digital infrastructure. Generalization is limited by the lack of longitudinal studies; multi-year student follow-ups are advised. By using multiple databases and institutional resources, the restriction on access to articles was lessened, guaranteeing a representative corpus.

On the other hand, each study was classified into three types of methodologies: mixed, quantitative and qualitative. Within the mixed approach, we distinguish that the Scopus and Web of Science databases have the highest representation with 31.58% and 60%, respectively. For the qualitative approach, we observed that ScienceDirect with 31.25% and Scopus with 21.05%. For the quantitative approach, it was observed that ScienceDirect is in the lead with 56.25% and Scopus with 47.37%. Fig. 7 shows the distribution of the studies mentioned at the beginning.

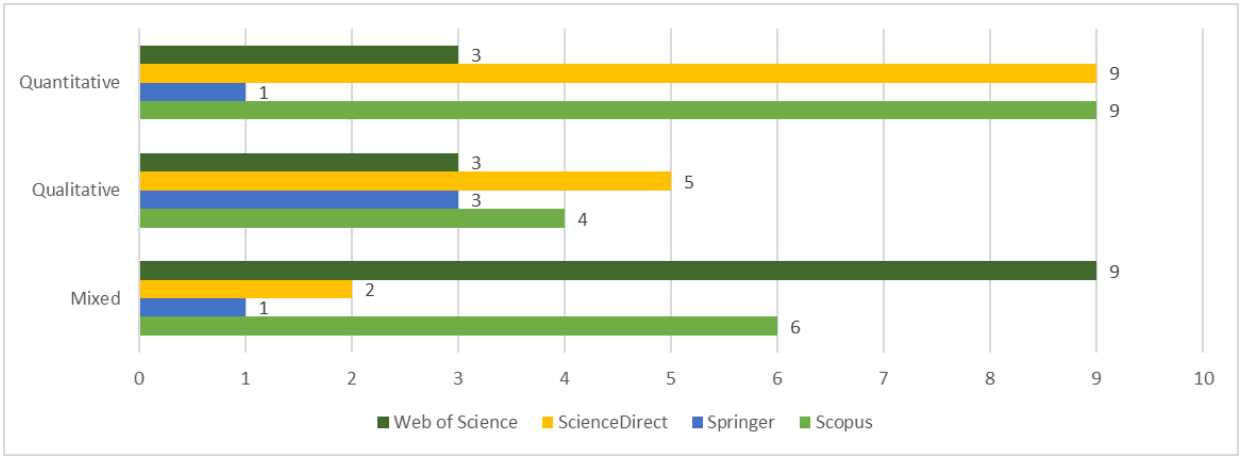


Fig. 7. Scientific bibliography incorporated into the research by methodological approach.

Finally, with the help of the "VOSviewer" tool, a graph was made showing the list of the most used terms in the research. Highlighting "chatbot" and "artificial intelligence" being the most concurrent with 27 and 19, respectively in the research, which shows how the focus of the articles is aligned to the research objectives. Among other keywords that are interconnected in the analysis, they show a different vision to the case study. The orange lines present the terms "innovation", "application", "design" and "learner", words that show how education has adopted these terms to improve its teaching methodologies. On the other hand, the purple lines show "age students", "university", "admission" and "issues",

where it is observed that mostly university students or students of different ages tend to use these tools to improve their learning. Finally, the blue lines, where "intelligent tutoring", "technology enhanced learning" and "educational chatbot" are mentioned, show how these technological tools are used as their own tutors or make them adopt teaching methodologies to benefit learning. Fig. 8 shows different groupings, where keywords are highlighted in a visual format that shows those areas identified in the studies reviewed.

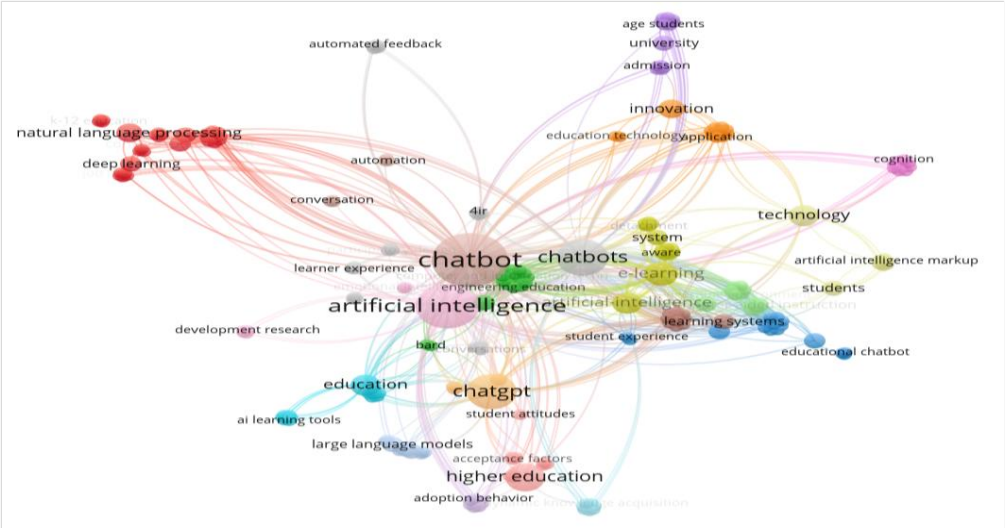


Fig. 8. Bibliometric exploration of the literature analyzed.

V. Discussion

This systematic study examines the integration of chatbots in e-learning environments, based on three inquiry questions. The main contributions are then presented and discussed in relation to the reference literature, highlighting convergences, discrepancies and relevant gaps that serve to delimit the current state of knowledge and guide future research.

A. Q1: What are the Benefits of Integrating Chatbots into Online Platforms?

The benefits obtained from the integration of chatbots in online learning environments note aspects of a pedagogical and operational nature. They emphasize dimensions such as personalization of learning, increased learner engagement, support for cognitive processes and optimization of educational efficiency. In this sense, the findings obtained are in line with the essential lines of research in the literature reviewed, in which tangible methodical evidence is provided.

The study [15] analyzed in depth the personalization of learning by devising an AI-based chatbot that employed NLP techniques to crawl multimedia content. The keyword extraction methodology allowed optimizing indexing, decreasing search times and increasing learner satisfaction. The results corroborate that the adaptability of the chatbots supports individualized learning paths, in agreement with the tabulated data on personalization.

Also, chatbots increase active student engagement, due to their immediate interactivity with the user. This claim is supported by research [16], as it analyzed public perception of ChatGPT in education, by web data mining and applying ML to a set of 2003 articles collected from the web, revealing identification supporting that chatbots activate educational institutions, while improving student competencies.

Regarding cognitive support, the MERLIN project, in [18], evaluated 102 Malaysian students with the use of a chatbot equipped with multimedia resources; this quantitative approach made it possible to verify the existence of progress at the cognitive level, proving that the timely provision of the assistance service contributes to improving conceptual assimilation processes.

Regarding educational efficiency, the findings are supported by research [21], in which 142 teachers with ESL were interviewed, using Likert-type surveys and the formulation of open questions, concluding that the use of chatbots simplifies feedback and reproduces the functioning of didactic interactions, pointing out the need for teacher training. Table II shows a summary of the distribution of the advantages identified.

TABLE II. Benefits of Integrating Chatbots into Online Platforms

#	Benefits	Quantity	References
1	Personalization of learning	13	[27], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37], [38], [39]
2	Student engagement	10	[40], [41], [42], [43], [44], [45], [46], [47], [48], [49]
3	Cognitive process support	7	[50], [51], [52], [53], [54], [55], [56]
4	Educational efficiency	7	[57], [58], [59], [60], [61], [62], [63]
5	Democratized access	7	[64], [65], [66], [67], [68], [69], [70]
6	Competency development	7	[71], [72], [73], [74], [75], [76], [77]
7	Pedagogical innovation	3	[78], [79], [80]

When going deeper into the rigorousness of the results in Table II, crucial nuances existing in the examined

complementary benefits, although less recurrent in the impact indexes, can be appreciated. Thus, the democratization of access, upheld by seven literary works [64], [65], [66], [67], [68], [69], [70], finds comparative support in [17], an investigation in which 360 Malaysian students made use of PPMH model chatbots to learn English. However, the simplicity of use removed the technical barriers, highlighting that the lack of social interaction in the use of chatbots limited their community acceptance, proving that, to democratize learning, comprehensive socio-technical approaches are needed.

The evolution of competencies reveals the need for deliberate pedagogical designs that support effective knowledge transfer. Thus, in [19], the authors investigated a group of 842 students using the diffusion of innovations theory, concluding that acceptance is driven by relative advantages and their compatibility, although usefulness does not always translate into improved competencies. In complementarity is the pedagogical innovation described by three studies [78], [79], [80], related to research [22], where a pilot computer networking program integrating ChatGPT with ILEs was conducted for three graduate students. The result indicated that a human-chatbot hybridization favors active methodologies, thus suggesting that innovation requires scalability to achieve widespread impact.

In summary, the conclusion of the present systematic review with respect to the body of literature treated as a precursor converges with the above, in the sense that chatbots positively transform online education. However, duality can be observed: while fundamental benefits such as personalization and engagement are the subject of strong consensus, complementary benefits related to access and innovation face conjunctural challenges that demand the implementation of adaptive strategies.

B. Q2: How does the Integration of Chatbots in e-learning Platforms Affect Academic Performance?

E-learning platforms integrated with chatbots have a significant impact on students' academic performance, thanks to transformational mechanisms capable of redefining the educational experience. According to the evidence obtained, the improvement of academic performance, the promotion of self-regulation and the increase of student participation are emerging as the predominant impacts. This synergetic relationship enables institutions to develop improved pedagogical processes based on intelligent support, combining innovation and educational effectiveness.

Research [20] provided crucial evidence on the improvement of academic performance, using a quantitative approach based on the comparison of part-time and full-time students. In this study, familiarization with ChatGPT and question formulation skills were analyzed, showing that the full-time group relied more on the linguistic model to stimulate their learning, which validates the tabular findings highlighting this impact in nine studies.

Regarding the promotion of self-regulation, research [23] explored knowledge management coefficients in Vietnamese university students. Applying a quantitative experimental approach, it was identified that validation of experiences reinforces perceived usefulness and self-regulation of learning, thus consolidating the metacognitive function of chatbots. These results support the seven tabular studies highlighting this benefit.

In relation to increased participation, research [24] evaluated the existence of ethical principles among students aged 12 to 24 years, using models of technological acceptance, and showed that chatbots encouraged constant interaction, particularly in the segment of adolescents aged 12 to 18 years, since these users repeatedly used them as support in academic decision making. The evidence confirms the six studies that highlight the potential of chatbots to stimulate student participation. Table III, given below, succinctly describes the magnitude of the impacts identified.

TABLE III. IMPACT ON ACADEMIC PERFORMANCE WHEN INTEGRATING CHATBOTS INTO E-LEARNING PLATFORMS

#	Impact	Quantity	References
1	Improves academic performance	9	[29], [31], [33], [37], [41], [43], [53], [74], [77]
2	Encourages self-regulation	7	[32], [38], [44], [50], [51], [52], [54]
3	Improves participation	6	[32], [40], [45], [46], [63], [79]
4	Optimizes formative assessment	5	[30], [34], [62], [75], [76]
5	Specific skills	5	[47], [48], [59], [73], [80]
6	Mixed or contextual effects	4	[36], [42], [49], [65]

On the other hand, the optimization of formative assessment, exposed by studies [30], [34], [62], [75], [76], denotes the streamlining of personalized feedback derived from chatbots, although research such as [19], based on the theory of diffusion of innovations, warns that the effectiveness of these agents is conditioned by the compatibility with institutional pedagogical designs.

Likewise, papers [47], [48], [59], [73], [80], note that the impact on specific skills is manifested improvement in technical or linguistic competencies because of specialized chatbots. This phenomenon was observed in [17] during the practice of English language learning, where the operational simplicity facilitated the acquisition of practical skills by using the PPHM model.

On the other hand, according to [36], [42], [49], [65], mixed or contextual effects reflected differences dependent on institutional factors. This duality is exposed in [20], where part-time students showed low academic performance, a fact that indicates that the impact depends on variables such as time spent and the educational conditions of the center.

C. Q3: What are the Technical and Pedagogical Challenges Faced by Educational Institutions when Implementing Chatbots in their e-Learning Platforms?

Educational institutions face obstacles linked to the integration of chatbots within e-learning platforms, whose technical barriers and pedagogical constraints constitute critical frictions. Evidence reveals that the major obstacles are related to NLP accuracy and adaptation to learning styles, as well as systemic integration issues that intensify teacher resistance.

The result reflects tensions between technological capabilities and real needs in teaching.

This problem is demonstrated in research [21], through an analysis conducted on a group of 142 ESL teachers, combining the use of Likert scales and qualitative exploration techniques. According to the results, NLP difficulties in contextual comprehension produce generic answers, without the possibility of personalization according to the different cognitive styles, a finding corroborated by nine tabular studies on this duality.

In parallel, the paper [19] applied the methodological paradigm of diffusion of innovations with 842 students, evidence in their study that the technical complexity of integrality requires unsustainable investments, which leaves teachers with the impression that chatbots intrude into their pedagogical autonomy. Table IV, given below, presents the challenges encountered after the rigorous analysis of the selected documents.

TABLE IV. TECHNICAL AND PEDAGOGICAL CHALLENGES IN EDUCATIONAL INSTITUTIONS AFTER IMPLEMENTING CHATBOTS IN E-LEARNING PLATFORMS

#	Technicians	Pedagogical	Quantity	References
1	Accuracy in NLP	Adaptation to learning styles	9	[27], [28], [29], [30], [32], [53], [58], [60], [62]
2	Content updating	Design of effective feedback	6	[41], [44], [57], [64], [68], [75]
3	Integration with platforms	Teacher resistance	7	[33], [35], [39], [46], [66], [67], [80]
4	Limitations in NLP	Lack of metacognitive strategies	6	[47], [51], [61], [63], [69], [81]
5	Complex queries	Competency assessment	5	[31], [34], [36], [65], [70]
6	Data privacy	Technology dependence	4	[37], [43], [56], [78]
7	Algorithmic biases	eacher training in AI	3	[54], [55], [71]
8	Infrastructure	Digital divide	3	[42], [45], [76]
9	Scalability	Autonomous learning	3	[52], [74], [79]
10	Latency in answers	Loss of human interaction	2	[48], [71]

As for the lower frequency, but high criticality challenges, literature boasts compelling arguments. Data privacy and technology dependency, present in four studies, [37], [43], [56], [78], are supported by [24], where technology acceptance models, applied to students aged 12 to 24, detected a wariness of personal data management, exacerbating vulnerabilities on an institutional scale.

Algorithmic biases in studies [54], [55], [71], are quantified by [16], by demonstrating, through their web mining analysis, the existence of a set of homogeneous linguistic texts capable of reproducing cultural biases linked to pedagogical responses. On the other hand, the digital divides, as found by [42], [45], [76], are exemplified by [17], since the Malaysian PPMH model showed that insufficient networks exclude 34% of the population in rural areas.

From the findings, a framework comprising four essential elements is derived: 1) technology and algorithms (NLP, personalization, data management); 2) education and training (active learning methods, training, curriculum); 3) students and learning (autoregulation, motivation, digital competencies); and 4) ethics and governance (privacy, privacy, institutional policies). The balanced interaction between these components shows that chatbot effectiveness depends on both technological innovation and pedagogical and esthetic support.

Finally, [22] validates the latency of responses, already noticed by two investigations [48], [71], since their experiment on computer networks showed a 40% reduction in the degree of confidence of students due to delays longer than ten seconds.

VI. CONCLUSION

SRL's research work analyzed different texts published between the period 2020 and 2024, with the purpose of examining the role of chatbots in e-learning platforms, evaluating their pedagogical benefits, their impact on academic performance and the technical and ethical challenges associated with their implementation. The collection of studies was made from different recognized databases, including Scopus, represented in 34.55% contributions, ScienceDirect with 29.09%, Web of Science with 27.27% and Springer in 9.09%, ensuring a diversified academic coverage.

The review identified that educational chatbots have a positive effect on the personalization of learning, student motivation, self-regulation and efficiency in educational processes. It also contributes significantly to the improvement of academic performance, especially when integrated in a coherent manner with active pedagogical strategies. There were also important difficulties associated with NLP: the need to adapt to different learning styles, the resistance of teachers to its use, and the limitations of the technological infrastructure in academic environments. Therefore, risks associated with data privacy, the emergence of algorithmic biases and the cost of interaction emerged, exposing the need for ethical frameworks to protect information and encourage teacher training regarding the use of tools in e-learning platforms.

Regarding the limitations of the study, the literature reviewed presents one of the typical shortcomings in this type of research: the scarcity of longitudinal studies and the limited availability of studies with terminology associated with the object of research “chatbots” and “e-learning”, and of those works that, although catalogued in the databases consulted, have not been accessible in full text, which has reduced the geographical scope and the subject of the corpus of text analyzed. Indeed, the corpus contains many reviewed works that rely on quantitative approaches and the use of self-reports without real context contrasts, limiting the studies to generalization.

However, and despite these limitations, the study provides a good basis for future research and at the same time serves as a synthesis of the most relevant findings in recent years on the impact of chatbots in digital education. The study invites researchers, developers and policy makers to deepen the analysis of the effectiveness of chatbots, the search for new

conditions of use and the design of new didactic, pedagogical and ethical strategies that allow to take advantage of the full potential offered by these tools to transform the teaching and student learning experience in online education.

REFERENCES

[1] Doh-Yeon Kim, "Reimagining education in Covid-19 - UNESCO Biblioteca Digital." Accessed: Apr. 06, 2025. [Online]. Available: https://unesdoc.unesco.org/ark:/48223/pf0000375556?posInSet=1&queryId=f26c0115-9337-403e-888c-b5dfa731eddc

[2] S. Sageengrana, S. Selvakumar, and S. Srinivasan, "Optimized RB-RNN: Development of hybrid deep learning for analyzing student's behaviours in online-learning using brain waves and chatbots," Expert Syst Appl, vol. 248, p. 123267, Aug. 2024, doi: 10.1016/j.eswa.2024.123267.

[3] M. M. A. Abdelmoiz, M. M. M. Mostafa, and T. H. A. Soliman, "Developing an Educational Chatbot for Scientific Data Management Course Using DialogFlow," Applied Mathematics & Information Sciences, vol. 18, no. 03, pp. 629–640, Mar. 2024, doi: 10.18576/amis/180315.

[4] A. Jain and P. Bhati, "Comparative Analysis and Development of Voice-based Chatbot System for Differently-abled," J Phys Conf Ser, vol. 2273, no. 1, p. 012003, Jan. 2022, doi: 10.1088/1742-6596/2273/1/012003.

[5] M. D. La Roca, M. M. Chan, A. Garcia-Cabot, E. Garcia-Lopez, and H. Amado-Salvatierra, "The impact of a chatbot working as an assistant in a course for supporting student learning and engagement," Computer Applications in Engineering Education, vol. 32, no. 5, p. e22750, Sep. 2024, doi: 10.1002/cae.22750.

[6] E. S. Elsawy, "The Reality of Using Cloud Computing in Information and Record Management in Oman," Journal of Ecohumanism, vol. 3, no. 4, pp. 2884–2898, Aug. 2024, doi: 10.62754/joe.v3i4.3804.

[7] C. H. Chuang, J. H. Lo, and Y. K. Wu, "Integrating Chatbot and Augmented Reality Technology into Biology Learning during COVID-19," Electronics (Switzerland), vol. 12, no. 1, p. 222, Jan. 2023, doi: 10.3390/electronics12010222.

[8] A. Wong, "The Design of an Intelligent Chatbot with Natural Language Processing Capabilities to Support Learners," J Phys Conf Ser, vol. 2251, no. 1, p. 012005, May 2022, doi: 10.1088/1742-6596/2251/1/012005.

[9] M. Gamboa-Ramos, R. Gómez-Noa, O. Iparraguirre-Villanueva, M. Cabanillas-Carbonell, and J. L. H. Salazar, "Mobile Application with Augmented Reality to Improve Learning in Science and Technology," International Journal of Advanced Computer Science and Applications, vol. 12, no. 10, pp. 487–492, 2021, doi: 10.14569/IJACSA.2021.0121055.

[10] G. Attigeri, A. Agrawal, and S. V. Kolekar, "Advanced NLP Models for Technical University Information Chatbots: Development and Comparative Analysis," IEEE Access, vol. 12, pp. 29633–29647, Jan. 2024, doi: 10.1109/ACCESS.2024.3368382.

[11] R. Sudheesh et al., "Analyzing Sentiments Regarding ChatGPT Using Novel BERT: A Machine Learning Approach," Information (Switzerland), vol. 14, no. 9, p. 474, Sep. 2023, doi: 10.3390/info14090474.

[12] H. Yang, "E-learning platforms in ideological and political education at universities: students' motivation and learning performance," BMC Med Educ, vol. 24, no. 1, p. 628, Dec. 2024, doi: 10.1186/s12909-024-05572-2.

[13] R. Malik, E. A. Sharma, S. Trivedi, and R. Mishra, "Adoption of Chatbots for Learning among University Students: Role of Perceived Convenience and Enhanced Performance," International Journal of Emerging Technologies in Learning, vol. 16, no. 18, pp. 200–212, Jan. 2021, doi: 10.3991/ijet.v16i18.24315.

[14] W. Kaiss, K. Mansouri, and F. Poirier, "Effectiveness of an Adaptive Learning Chatbot on Students' Learning Outcomes Based on Learning Styles," International Journal of Emerging Technologies in Learning, vol. 18, no. 13, pp. 250–261, Jan. 2023, doi: 10.3991/ijet.v18i13.39329.

[15] S. El Janati, A. Maach, and D. El Ghanami, "Adaptive e-learning AI-powered chatbot based on multimedia indexing," International Journal of Advanced Computer Science and Applications, vol. 11, no. 12, pp. 299–308, Dec. 2020, doi: 10.14569/IJACSA.2020.0111238.

[16] A. Rejeb, K. Rejeb, A. Appolloni, H. Treiblmaier, and M. Iranmanesh, "Exploring the impact of ChatGPT on education: A web mining and machine learning approach," The International Journal of Management Education, vol. 22, no. 1, p. 100932, Mar. 2024, doi: 10.1016/J.IJME.2024.100932.

[17] N. Annamalai, R. A. Rashid, U. Munir Hashmi, M. Mohamed, M. Harb Alqaryouti, and A. Eddin Sadeq, "Using chatbots for English language learning in higher education," Computers and Education: Artificial Intelligence, vol. 5, p. 100153, Jan. 2023, doi: 10.1016/J.CAEAI.2023.100153.

[18] M. Neo, "THE MERLIN PROJECT: MALAYSIAN STUDENTS' ACCEPTANCE OF AN AI CHATBOT IN THEIR LEARNING PROCESS," Turkish Online Journal of Distance Education, vol. 23, no. 3, pp. 31–48, Jan. 2022, doi: 10.17718/tojde.1137122.

[19] M. A. Ayanwale and M. Ndlovu, "Investigating factors of students' behavioral intentions to adopt chatbot technologies in higher education: Perspective from expanded diffusion theory of innovation," Computers in Human Behavior Reports, vol. 14, p. 100396, May 2024, doi: 10.1016/J.CHBR.2024.100396.

[20] R. Niedbal, A. Sokolowski, and A. Wrzalik, "Students' Use of the Artificial Intelligence Language Model in their Learning Process," Procedia Comput Sci, vol. 225, pp. 3059–3066, Jan. 2023, doi: 10.1016/J.PROCS.2023.10.299.

[21] K. M. Chuah and M. K. Kabilan, "Teachers' Views on the Use of Chatbots to Support English Language Teaching in a Mobile Environment," International Journal of Emerging Technologies in Learning, vol. 16, no. 20, pp. 223–237, Jan. 2021, doi: 10.3991/ijet.v16i20.24917.

[22] D. Soto, M. Higashida, S. Shirai, M. Ueda, and Y. Uranishi, "Enhancing Learning Dynamics: Integrating Interactive Learning Environments and ChatGPT for Computer Networking Lessons," Procedia Comput Sci, vol. 246, no. C, pp. 3595–3604, Jan. 2024, doi: 10.1016/J.PROCS.2024.09.198.

[23] T. T. A. Ngo, T. T. Tran, G. K. An, and P. T. Nguyen, "ChatGPT for Educational Purposes: Investigating the Impact of Knowledge Management Factors on Student Satisfaction and Continuous Usage," IEEE Transactions on Learning Technologies, vol. 17, pp. 1367–1378, Jan. 2024, doi: 10.1109/TLT.2024.3383773.

[24] Y. Kajiwara and K. Kawabata, "AI literacy for ethical use of chatbot: Will students accept AI ethics?," Computers and Education: Artificial Intelligence, vol. 6, p. 100251, Jun. 2024, doi: 10.1016/J.CAEAI.2024.100251.

[25] D. Moher, A. Liberati, J. Tetzlaff, and D. G. Altman, "Preferred reporting items for systematic reviews and meta-analyses: The PRISMA statement," International Journal of Surgery, vol. 8, no. 5, pp. 336–341, Jan. 2010, doi: 10.1016/J.IJSU.2010.02.007.

[26] N. R. Haddaway, M. J. Page, C. C. Pritchard, and L. A. McGuinness, "PRISMA2020: An R package and Shiny app for producing PRISMA 2020-compliant flow diagrams, with interactivity for optimised digital transparency and Open Synthesis," Campbell Systematic Reviews, vol. 18, no. 2, p. e1230, Jun. 2022, doi: 10.1002/CL2.1230;WGROUP:STRING:PUBLICATION.

[27] A. Raza, M. Latif, M. Umer Farooq, M. Adnan Baig, and M. Ali Akhtar, "Enabling Context-based AI in Chatbots for conveying Personalized Interdisciplinary Knowledge to Users," Engineering, Technology and Applied Science Research, vol. 13, no. 6, pp. 12231–12236, 2023, doi: 10.48084/etasr.6313.

[28] A. Alabbas and K. Alomar, "Tayseer: A Novel AI-Powered Arabic Chatbot Framework for Technical and Vocational Student Helpdesk Services and Enhancing Student Interactions," Applied Sciences (Switzerland), vol. 14, no. 6, 2024, doi: 10.3390/app14062547.

[29] C. B. Yao and Y. L. Wu, "Intelligent and Interactive Chatbot Based on the Recommendation Mechanism to Reach Personalized Learning," INTERNATIONAL JOURNAL OF INFORMATION AND COMMUNICATION TECHNOLOGY EDUCATION, vol. 18, no. 1, 2022, doi: 10.4018/IJICTE.315596.

[30] K. Hallal, R. Hamdan, and S. Tlais, "Exploring the potential of AI-Chatbots in organic chemistry: An assessment of ChatGPT and Bard," Computers and Education: Artificial Intelligence, vol. 5, p. 100170, 2023, doi: https://doi.org/10.1016/j.caeai.2023.100170.

[31] J. A. Kumar, "Educational chatbots for project-based learning: investigating learning outcomes for a team-based design course," INTERNATIONAL JOURNAL OF EDUCATIONAL TECHNOLOGY IN HIGHER EDUCATION, vol. 18, no. 1, 2021, doi: 10.1186/s41239-021-00302-w.

[32] W. Villegas-Ch, A. Arias-Navarrete, and X. Palacios-Pacheco, "Proposal of an Architecture for the Integration of a Chatbot with Artificial Intelligence in a Smart Campus for the Improvement of Learning," Sustainability, vol. 12, no. 4, 2020, doi: 10.3390/su12041500.

[33] T. Adiguzel, M. H. Kaya, and F. K. Cansu, "Revolutionizing education with AI: Exploring the transformative potential of ChatGPT," Contemp Educ Technol, vol. 15, no. 3, 2023, doi: 10.30935/cedtech/13152.

[34] A. Carrasco Rodríguez, "Reinventing the Teaching of Early Modern History in Secondary School: the use of ChatGPT to Enhance Learning and Educational Innovation," Studia Historica: Historia Moderna, vol. 45, no. 1, pp. 101–145, Jul. 2023, doi: 10.14201/shhmo2023451101146.

[35] S. Ji and T. Yuan, "Conversational Intelligent Tutoring Systems for Online Learning: What do Students and Tutors Say?," in IEEE Global Engineering Education Conference, EDUCON, M. Jemni, I. Kallel, and A. Akkari, Eds., IEEE Computer Society, 2022, pp. 292–298. doi: 10.1109/EDUCON52537.2022.9766567.

[36] P. Chatwattana, P. Yangthisarn, and A. Tabubpha, "The Educational Recommendation System with Artificial Intelligence Chatbot: A Case Study in Thailan," International Journal of Engineering Pedagogy, vol. 14, no. 5, pp. 51–64, 2024, doi: 10.3991/ijep.v14i5.48491.

[37] B. Kayali, M. Yavuz, S. Balat, and M. Calisan, "Investigation of student experiences with ChatGPT-supported online learning applications in higher education," AUSTRALASIAN JOURNAL OF EDUCATIONAL TECHNOLOGY, vol. 39, no. 5, pp. 20–39, 2023, doi: 10.14742/ajet.8915.

[38] C. Karrenbauer, T. Brauner, C. M. König, and M. H. Breitner, "Design, development, and evaluation of an individual digital study assistant for higher education students," Educational Technology Research and Development, vol. 71, no. 5, pp. 2047–2071, Oct. 2023, doi: 10.1007/S11423-023-10255-8.

[39] A. S. Almogren, W. M. Al-Rahmi, and N. A. Dahri, "Exploring factors influencing the acceptance of ChatGPT in higher education: A smart education perspective," Heliyon, vol. 10, no. 11, p. e31887, 2024, doi: https://doi.org/10.1016/j.heliyon.2024.e31887.

[40] M. Neo et al., "Enhancing Students' Online Learning Experiences with Artificial Intelligence (AI): The MERLIN Project," International Journal of Technology, vol. 13, no. 5, pp. 1023–1034, 2022, doi: 10.14716/ijtech.v13i5.5843.

[41] E. Ortega-Ochoa, J. Quiroga Pérez, M. Arguedas, T. Daradoumis, and J. M. Marquès Puig, "The effectiveness of empathic chatbot feedback for developing computer competencies, motivation, self-regulation, and metacognitive reasoning in online higher education," Internet of Things, vol. 25, p. 101101, 2024, doi: https://doi.org/10.1016/j.iot.2024.101101.

[42] C. Y. Lai, K. Y. Cheung, and C. S. Chan, "Exploring the role of intrinsic motivation in ChatGPT adoption to support active learning: An extension of the technology acceptance model," Computers and Education: Artificial Intelligence, vol. 5, p. 100178, 2023, doi: https://doi.org/10.1016/j.caeai.2023.100178.

[43] H. Jo, "From concerns to benefits: a comprehensive study of ChatGPT usage in education," INTERNATIONAL JOURNAL OF EDUCATIONAL TECHNOLOGY IN HIGHER EDUCATION, vol. 21, no. 1, 2024, doi: 10.1186/s41239-024-00471-4.

[44] M. C. Sáiz-Manzanares, R. Marticorena-Sánchez, L. J. Martín-Antón, I. González Díez, and L. Almeida, "Perceived satisfaction of university students with the use of chatbots as a tool for self-regulated learning," Heliyon, vol. 9, no. 1, p. e12843, 2023, doi: https://doi.org/10.1016/j.heliyon.2023.e12843.

[45] E. H. K. Wu, C. H. Lin, Y. Y. Ou, C. Z. Liu, W. K. Wang, and C. Y. Chao, "Advantages and Constraints of a Hybrid Model K-12 E-Learning Assistant Chatbot," IEEE ACCESS, vol. 8, pp. 77788–77801, 2020, doi: 10.1109/ACCESS.2020.2988252.

[46] N. Abbas, J. Whitfield, E. Atwell, H. Bowman, T. Pickard, and A. Walker, "Online chat and chatbots to enhance mature student engagement in higher education," INTERNATIONAL JOURNAL OF LIFELONG EDUCATION, vol. 41, no. 3, pp. 308–326, 2022, doi: 10.1080/02601370.2022.2066213.

[47] D. Jackson and A. Latham, "Talk to The Ghost: The Storybox methodology for faster development of storytelling chatbots," Expert Syst Appl, vol. 190, 2022, doi: 10.1016/j.eswa.2021.116223.

[48] S. M. Mosleh, F. A. Alsaadi, F. K. Alnaqbi, M. A. Alkhzaimi, S. W. Alnaqbi, and W. M. Alsereidi, "Examining the association between emotional intelligence and chatbot utilization in education: A cross-sectional examination of undergraduate students in the UAE," Heliyon, vol. 10, no. 11, p. e31952, 2024, doi: https://doi.org/10.1016/j.heliyon.2024.e31952.

[49] N. Hasan, J. A. Polin, M. R. Ahmmed, M. M. Sakib, M. F. Jahin, and M. M. Rahman, "A novel approach to analyzing the impact of AI, ChatGPT, and chatbot on education using machine learning algorithms," Bulletin of Electrical Engineering and Informatics, vol. 13, no. 4, pp. 2951–2958, 2024, doi: 10.11591/eei.v13i4.7158.

[50] W. Kaiss, K. Mansouri, and F. Poirier, "Chatbot Design to Help Learners Self-Regulte Their Learning in Online Learning Environments," in Proceedings - 2023 IEEE International Conference on Advanced Learning Technologies, ICALT 2023, M. Chang, C. N.-S., R. Kuo, G. Rudolph, D. G. Sampson, and A. Tlili, Eds., Institute of Electrical and Electronics Engineers Inc., 2023, pp. 236–238. doi: 10.1109/ICALT58122.2023.00075.

[51] L. Martins, M. Fernández-Ferrer, and E. Puertas, "Analysing Conversation Pathways with a Chatbot Tutor to Enhance Self-Regulation in Higher Education," Educ Sci (Basel), vol. 14, no. 6, 2024, doi: 10.3390/educsci14060590.

[52] A. T. Neumann et al., "Chatbots as a Tool to Scale Mentoring Processes: Individually Supporting Self-Study in Higher Education," Front Artif Intell, vol. 4, 2021, doi: 10.3389/frai.2021.668220.

[53] W. M. A. F. Wan Hamzah, I. Ismail, M. K. Yusof, S. I. M. Saany, and A. Yacob, "Using Learning Analytics to Explore Responses from Student Conversations with Chatbot for Education," International Journal of Engineering Pedagogy, vol. 11, no. 6, pp. 70–84, 2021, doi: 10.3991/ijep.v11i6.23475.

[54] S. M. Araujo and R. Cruz-Correia, "Incorporating ChatGPT in Medical Informatics Education: Mixed Methods Study on Student Perceptions and Experiential Integration Proposals," JMIR Med Educ, vol. 10, 2024, doi: 10.2196/51151.

[55] S. A. Chauncey and H. P. McKenna, "A framework and exemplars for ethical and responsible use of AI Chatbot technology to support teaching and learning," Computers and Education: Artificial Intelligence, vol. 5, p. 100182, 2023, doi: https://doi.org/10.1016/j.caeai.2023.100182.

[56] Y. Kajiwara and K. Kawabata, "AI literacy for ethical use of chatbot: Will students accept AI ethics?," Computers and Education: Artificial Intelligence, vol. 6, p. 100251, 2024, doi: https://doi.org/10.1016/j.caeai.2024.100251.

[57] S. C. Man, O. Matei, T. Faragau, L. Andreica, and D. Daraba, "The Innovative Use of Intelligent Chatbot for Sustainable Health Education Admission Process: Learnt Lessons and Good Practices," Applied Sciences (Switzerland), vol. 13, no. 4, 2023, doi: 10.3390/app13042415.

[58] N. A. Said, D. Gura, and D. Karlov, "Efficiency of Smart AI-Based Voice Apps and Virtual Services Operating With Chatbots," Mendel, vol. 28, no. 2, pp. 1–8, 2022, doi: 10.13164/mendel.2022.2.009.

[59] S. H. Anwar, K. M. Abouaish, E. M. Matta, A. K. Farouq, A. A. Ahmed, and N. K. Negied, "Academic assistance chatbot-a comprehensive NLP and deep learning-based approaches," Indonesian Journal of Electrical Engineering and Computer Science, vol. 33, no. 2, pp. 1042–1056, 2024, doi: 10.11591/ijeecs.v33.i2.pp1042-1056.

[60] K. Sridharan and R. P. Sequeira, "Artificial intelligence and medical education: application in classroom instruction and student assessment using a pharmacology & therapeutics case study," BMC Med Educ, vol. 24, no. 1, Dec. 2024, doi: 10.1186/S12909-024-05365-7.

[61] N. I. Mohd Rahim, N. A. Iahad, A. F. Yusof, and M. A. Al-Sharafi, "AI-Based Chatbots Adoption Model for Higher-Education Institutions: A Hybrid PLS-SEM-Neural Network Modelling Approach," Sustainability (Switzerland), vol. 14, no. 19, 2022, doi: 10.3390/su141912726.

[62] O. Kolade, A. Owoseni, and A. Egbetokun, "Is AI changing learning and assessment as we know it? Evidence from a ChatGPT experiment and a conceptual framework," Heliyon, vol. 10, no. 4, p. e25953, 2024, doi: https://doi.org/10.1016/j.heliyon.2024.e25953.

[63] Y. H. Chien and C. K. Yao, "Development of an AI Userbot for Engineering Design Education Using an Intent and Flow Combined Framework," APPLIED SCIENCES-BASEL, vol. 10, no. 22, 2020, doi: 10.3390/app10227970.

[64] K. Mzwri and M. Turcsányi-Szabo, "Internet Wizard for Enhancing Open-Domain Question-Answering Chatbot Knowledge Base in Education," Applied Sciences (Switzerland), vol. 13, no. 14, 2023, doi: 10.3390/app13148114.

[65] N. Alfirevic, D. G. Pranicevic, and M. Mabic, "Custom-Trained Large Language Models as Open Educational Resources: An Exploratory Research of a Business Management Educational Chatbot in Croatia and Bosnia and Herzegovina," Sustainability, vol. 16, no. 12, 2024, doi: 10.3390/su16124929.

[66] M. A. Ayanwale and M. Ndlovu, "Investigating factors of students' behavioral intentions to adopt chatbot technologies in higher education: Perspective from expanded diffusion theory of innovation," Computers in Human Behavior Reports, vol. 14, p. 100396, 2024, doi: https://doi.org/10.1016/j.chbr.2024.100396.

[67] R. Mash, D. Schouw, and A. E. Fischer, "Evaluating the Implementation of the GREAT4Diabetes WhatsApp Chatbot to Educate People with Type 2 Diabetes during the COVID-19 Pandemic: Convergent Mixed Methods Study," JMIR Diabetes, vol. 7, no. 2, 2022, doi: 10.2196/37882.

[68] J. S. Corrêa, A. P. D. A. Neto, G. R. Pinto, L. D. B. Lima, and A. S. Teles, "Lhia: A Smart Chatbot for Breastfeeding Education and Recruitment of Human Milk Donors," Applied Sciences (Switzerland), vol. 13, no. 12, 2023, doi: 10.3390/app13126923.

[69] D. Lee, H. Kim, and S.-H. Sung, "Development research on an AI English learning support system to facilitate learner-generated-context-based learning," Educational technology research and development, vol. 71, no. 2, pp. 629–666, Apr. 2023, doi: 10.1007/s11423-022-10172-2.

[70] T. S. Sathe, J. Roshal, A. Naaseh, J. C. L'Huillier, S. M. Navarro, and C. Silvestri, "How I GPT It: Development of Custom Artificial Intelligence (AI) Chatbots for Surgical Education," J Surg Educ, vol. 81, no. 6, pp. 772–775, 2024, doi: https://doi.org/10.1016/j.jsurg.2024.03.004.

[71] L. Naamati-Schneider, "Enhancing AI competence in health management: students' experiences with ChatGPT as a learning Tool," BMC Med Educ, vol. 24, no. 1, Dec. 2024, doi: 10.1186/S12909-024-05595-9.

[72] A. Strzelecki, "Students' Acceptance of ChatGPT in Higher Education: An Extended Unified Theory of Acceptance and Use of Technology," Innov High Educ, vol. 49, no. 2, pp. 223–245, Apr. 2024, doi: 10.1007/S10755-023-09686-1.

[73] Y. Park and Y. Shin, "A Block-Based Interactive Programming Environment for Large-Scale Machine Learning Education," Applied Sciences (Switzerland), vol. 12, no. 24, 2022, doi: 10.3390/app122413008.

[74] Z. Zhang and X. Huang, "The impact of chatbots based on large language models on second language vocabulary acquisition," Heliyon, vol. 10, no. 3, p. e25370, 2024, doi: https://doi.org/10.1016/j.heliyon.2024.e25370.

[75] S. N. Yildirim-Erbasli and O. Bulut, "Conversation-based assessment: A novel approach to boosting test-taking effort in digital formative assessment," Computers and Education: Artificial Intelligence, vol. 4, p. 100135, 2023, doi: https://doi.org/10.1016/j.caeai.2023.100135.

[76] C. Stöhr, A. W. Ou, and H. Malmström, "Perceptions and usage of AI chatbots among students in higher education across genders, academic levels and fields of study," Computers and Education: Artificial Intelligence, vol. 7, p. 100259, 2024, doi: https://doi.org/10.1016/j.caeai.2024.100259.

[77] R. Mellado-Silva, A. Faúndez-Ugalde, and M. Blanco-Lobos, "Effective learning of tax regulations using different chatbot techniques," Advances in Science, Technology and Engineering Systems, vol. 5, no. 6, pp. 439–446, 2020, doi: 10.25046/aj050652.

[78] M. Van Poucke, "ChatGPT, the perfect virtual teaching assistant? Ideological bias in learner-chatbot interactions," Comput Compos, vol. 73, p. 102871, 2024, doi: https://doi.org/10.1016/j.compcom.2024.102871.

[79] Y. C. Chien, T. T. Wu, C. H. Lai, and Y. M. Huang, "Investigation of the Influence of Artificial Intelligence Markup Language-Based LINE ChatBot in Contextual English Learning," Front Psychol, vol. 13, 2022, doi: 10.3389/fpsyg.2022.785752.

[80] T.-C. Yang and J.-H. Chen, "Pre-service teachers' perceptions and intentions regarding the use of chatbots through statistical and lag sequential analysis," Computers and Education: Artificial Intelligence, vol. 4, p. 100119, 2023, doi: https://doi.org/10.1016/j.caeai.2022.100119.

[81] J. Henderson et al., "Development of a Bespoke Chatbot Design Tool to Facilitate a Crowd-based Co-creation Process," in 2022 International Conference on Interactive Media, Smart Systems and Emerging Technologies, IMET 2022 - Proceedings, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/IMET54801.2022.9929752.

A Hybrid Approach Combining Deep CNN Features with Classical Machine Learning for Diabetic Retinopathy Diagnosis

Amandeep Kaur[1], Simranjit Singh[2], Hardeep Singh[3]*, Sarveshwar Bharti[4], Jai Sharma[5], Himanshi Sharma[6]
Department of Computer Engineering and Technology, Guru Nanak Dev University, Amritsar, 143005, India[1, 2, 3, 5, 6]
Department of Computer Science, Guru Nanak Dev University, Amritsar, 143005, India[4]

***Abstract*—One of the main causes of vision impairment is diabetic retinopathy (DR), a common and dangerous consequence of diabetes that damages the retinal blood vessels. Preventing irreversible vision loss requires early detection of DR. Recent developments demonstrate how artificial intelligence (AI), and in particular deep learning (DL), can automate the classification of retinal images for the diagnosis of DR. In this study, a hybrid model is proposed that combines deep learning-based feature extraction with classical machine learning classifiers for robust medical image analysis. After using preprocessing methods to lower background noise, this study investigates the use of Convolutional Neural Networks (CNNs) for extracting discriminative features from DR images. To improve image contrast and highlight vascular features, the preprocessing pipeline uses morphological top-hat filtering and green channel extraction. Furthermore, transfer learning was applied to enhance feature representation. The tuned Radial Basis Function Support Vector Machine (RBF-SVM) had the greatest classification accuracy of 85% among the machine learning (ML) classifiers that were assessed, including Random Forest (RF), Gradient Boosting (GB), and RBF-SVM. These findings demonstrate the potential of hybrid AI-driven approaches and domain-specific medical image analysis in providing reliable and efficient automated DR detection.**

***Keywords*—Deep learning; convolutional neural networks; hybrid model; diabetic retinopathy; machine learning; medical image analysis; feature extraction**

I. Introduction

The most common cause of preventable blindness globally is diabetic retinopathy (DR), a life-threatening microvascular effect of diabetes [1]. Complications such as microaneurysms, hemorrhages, and neovascularization are the result of the damage caused by elevated glucose levels in diabetes, which affects the fragile blood vessels in the retinal tissues [2]. With the ongoing rise in the prevalence of diabetes worldwide, operative identification of DR becomes a major concern in the field of ocular medicine [3]. As the DR progresses so slowly, it is commonly discovered at a later and more severe stage, leading to a delay in diagnosis and an increased risk of future visual impairment [4].

Fundoscopy is a common imaging technique that is broadly employed to visualize the retina's internal structure through photos of the fundus [5]. Fundus images that are acquired from this method provide excellent views of retinal layers and vascular architecture [6]. However, the inspection of these images manually by an ophthalmologist not only requires plenty of time but also highly depends on the ability of the clinician. According to different studies, such as Benbassat and Pratt et al. [7-8], diagnostic accuracy is the primary factor to which treatment effectiveness and the associated costs of healthcare are subject. An intervention is effective if the DR is detected in its early stage.

Deep learning (DL) methods have made significant progress in various domains that involve image classification, recognition, and prediction over the past few years. Al Ayoubi et al. [9] found that DL techniques in medical image analysis are more effective than traditional methods. Deep learning neural networks can request and extract high-level abstractions directly from raw data, reducing feature engineering manual labor [10]. Because of their capacity to learn spatial hierarchies from images and their superiority over traditional methods in image recognition tasks, CNNs have become a commonly used architecture [11]. Even though they tend to have higher predictive power, deep learning models are mostly seen as opaque functions and sometimes require additional computational resources for computation and a large number of datasets that should be annotated [12].

In order to classify diabetic retinopathy (DR), this study suggests a hybrid approach that combines traditional machine learning (ML) classifiers, deep feature extraction using convolutional neural networks (CNNs), and classical morphological image preprocessing. Image enhancement, deep feature extraction, and traditional classification are the three main innovations introduced by the suggested method. Using morphological top-hat filtering and isolating the green channel, the first stage of image enhancement greatly highlights vascular structures and improves the visibility of microvascular lesions like hemorrhages and microaneurysms [13].

Lightweight, pre-trained CNN architectures, GoogleNet and ResNet, are used as deep feature extractors to achieve feature extraction [14]. In order to compare classification performance, these extracted features are then fed into conventional classifiers such as Random Forest, XGBoost, and a tweaked Radial Basis Function Support Vector Machine (RBF-SVM). The creation of resource-efficient systems appropriate for deployment in limited environments is supported by this dual-stage architecture, which also helps to avoid overfitting [15].

*Corresponding Author, email: - hardeep.cet@gndu.ac.in

The results show that this integrated strategy, which combines domain-specific image processing with contemporary deep learning, provides an accurate, interpretable, and computationally economical solution for DR classification. Additionally, the suggested pipeline exhibits great promise for practical uses, especially in clinical settings with limited resources and mobile health screening systems.

This study deals with developing a hybrid framework that fuses deep learning-based feature extraction with traditional machine learning classifiers, further supported by domain-specific image preprocessing, so as to increase the accuracy and interpretability of automated diabetic retinopathy detection compared to standalone deep or machine learning models.

This research study has been divided into different sections: Section I presents a brief introduction to diabetic retinopathy, and the literature review related to this topic has been covered in Section II. Section III describes the methodology followed in the research problem, while Section IV presents the different results from various algorithms. Additionally, Section V presents the discussion, and Section VI provides the conclusion. Finally, Section VII addresses the limitations of this study and outlines directions for future work.

II. Literature Review

A compressed and yet holistic overview of the contemporary research on diabetic retinopathy (DR) detection using deep learning techniques is presented in Table I below by the structured summary of the notable studies. Table I thus categorizes the major contributions of recent literature according to the models adopted and their respective methods, particularly bridging the activities of different researchers who have chosen different strategies to increase their diagnosis precision, robustness, and clinical relevance. Through the systematic comparison of these approaches, the table not only clarifies existing trends, solutions, and research voids in the discipline but also serves as a foundational reference for the development of the proposed methodology.

TABLE I. Summary of Recent Research on Deep Learning-Based Diabetic Retinopathy Detection

S. No.	Author(s) and Year	Methodology / Model Used	Key Contribution / Findings
1	Cao et al. (2024) [17]	Hybrid model combining Vision Transformers and CNNs	Enhanced feature extraction and classification via a hybrid architecture
2	Luo et al. (2024) [18]	CNN with local and long-range dependency modeling	Improved classification by capturing both short and long-term spatial patterns
3	Khaparde et al. (2023) [19]	Attention-based Swin U-Net (hybrid DL architecture)	Integrated segmentation and classification to enhance diagnostic accuracy
4	Li et al. (2022) [20]	Lesion-attention pyramid network	Improved DR grading by focusing on lesion-specific features
5	Bala et al. (2024) [21]	Comparative study between DL and conventional ML models	Deep learning outperformed ML, emphasizing the importance of model selection
6	Akella & Kumar (2024) [22]	Optimized deep learning using color fundus images	Focused on model optimization and preprocessing to improve DR classification
7	Saranya et al. (2023) [23]	CNN-based red lesion detection and classification	Demonstrated high accuracy in identifying small red lesions related to DR
8	Saini et al. (2023) [24]	CNN on OCT images for lesion prediction	Effective for diagnosing DR and macular oedema
9	Agarwal & Bhat (2023) [25]	Review of DL advances in DR diagnosis	Emphasized the role of DL in early detection and future potential
10	Dubey & Dixit (2023) [26]	Review of DL-based decision support systems	Highlighted need for reliable datasets and standard evaluation metrics
11	Vijayan & Salim (2023) [27]	Survey on DL-based automated DR systems	Identified clinically applicable solutions and grading mechanisms
12	Atwany et al. (2022) [28]	Comparative study of DL techniques	Presented performance metrics and evaluation of DR detection models
13	Farooq et al. (2022) [29]	CAD systems using DL for DR screening	Discussed clinical utility and integration in large-scale DR screening
14	Bidwai et al. (2022) [30]	Systematic review on ML trust and data quality	Stressed need for validated datasets and trustworthy models
15	Lalithadevi& Krishnaveni (2022) [31]	DL and image processing for retinal disease detection	Summarized clinical applications and technological advances in DR diagnosis
16	Dayana & Emmanuel (2023) [32]	Deep learning with metaheuristic optimization	Proposed a future-ready framework for DR screening using adaptive optimization

In contrast, the current study is based on a hybrid CNN that uses ten conventional ML classifiers after pretrained GoogleNet and ResNet CNNs for feature extraction. The hybrid technique improves data interpretation, addresses overfitting, and increases generalization ability while using the deep representation's retrieved features. The main objectives of this study are as follows:

- Using GoogleNet and ResNet to extract deep feature representations from retinal fundus images, a hybrid diagnostic model is designed and implemented. These features are then provided as input to a collection of conventional ML classifiers.
- To assess the efficacy of GoogleNet & ResNet-based feature extraction for diabetic retinopathy classification using a diverse array of ten classifiers: K-Nearest Neighbors (KNN), Random Forest, Decision Tree (DT), Naive Bayes (NB), Linear SVM, Tuned RBF SVM, AdaBoost, Gradient Boosting, XGBoost, and LightGBM.

- Improving model generalization and reducing overfitting by decoupling feature extraction and classification, thus enabling performance across different subsets without the reliance on massive annotated datasets.
- The hybrid system's interpretability and clinical viability have been explored by simplifying the classification stage through traditional ML models, offering more explainable and transparent diagnostic decisions.
- Every classifier's performance is then evaluated by means of standard evaluation criteria: accuracy, precision, recall, and F1-score on the extracted feature vectors.

III. Methodology

A. Dataset Employed

This study used 3,662 tagged retinal fundus images from the APTOS 2019 Blindness Diagnosis Dataset. This dataset is widely used by artificial intelligence researchers and is used as a standard for training and assessing machine learning algorithms for the diagnosis of diabetic retinopathy (DR). Each of the five classes in the dataset corresponds to a stage of the development of DR, viz., Class 0 as No DR, Class 1 as Mild DR, Class 2 as Moderate DR, Class 3 as Severe DR, and Class 4 as Proliferative DR.

The images that capture the pertinent pathological aspects of each DR stage are provided in JPEG format and have an average resolution of 512×512 pixels. Table II provides a thorough explanation of the dataset. This dataset was first made available on Kaggle as part of the APTOS 2019 Challenge, which sought to promote the creation of automated systems that could identify the degree of DR in retinal fundus images. Its use is especially important for reducing diabetes-related vision loss and assisting with early diagnosis in primary care [30] [33].

TABLE II. Dataset Description

Category	Details
Dataset Name	APTOS 2019 Blindness Detection Dataset
Dataset Size	3,662 images
Link	APTOS 2019 Dataset on Kaggle
File Type	PNG images
Author	APTOS 2019 Challenge, Kaggle
Purpose	To develop models that automatically detect DR and classify its severity levels. The dataset is used to train and evaluate algorithms in the field of DR detection.
Number of Images	3,662 images
Image Types	5-class classification: 0 (No DR), 1 (Mild DR), 2 (Moderate DR), 3 (Severe DR), 4 (Proliferative DR)
Image Format	PNG (each image is a fundus retinal image, labeled with severity level)
Resolution	Images vary in resolution but are typically around 512x512 pixels.

B. Data Preprocessing

Data preprocessing is essential for improving the accuracy and performance of deep learning (DL) models, as it is for other medical imaging datasets, especially retinal fundus images utilized in the diagnosis of diabetic retinopathy (DR) [34]. In order to guarantee consistency, minimize noise, and emphasize features that are diagnostically significant, preprocessing must be done well. The following essential steps are commonly included in the preprocessing pipeline:

1) Image resizing: To change the default, adjust the template as follows. Scaling as a requirement can help standardize all retinal fundus images into the same size because they vary in size [34]. In this instance, the consistent paradigm of image handling is combined with a reduction in computational effort. To ensure consistency in the dataset and shorten the model's training time, it is standard procedure to resize each image to a precise size, which involves 224x224 pixels, as shown in Fig. 1.

2) Green channel extraction: To change the default, adjust the template as follows. The green channel in retinal fundus imaging is remarkable at providing important information, and thus, it is the one that focuses on the blood vessels and the lesions in the most effective way. By utilizing only the green channel to remove the dimensionality, the preprocessing phase becomes simpler and at the same time achieves the goal of retaining those features that are necessary to determine diabetic retinopathy, as seen in Fig. 1. Another method of improving the model's capacity to represent the markers that are indicative of the evolving disease is the targeted channel extraction, which involves channeling out less important color information from the red and blue channels [35].

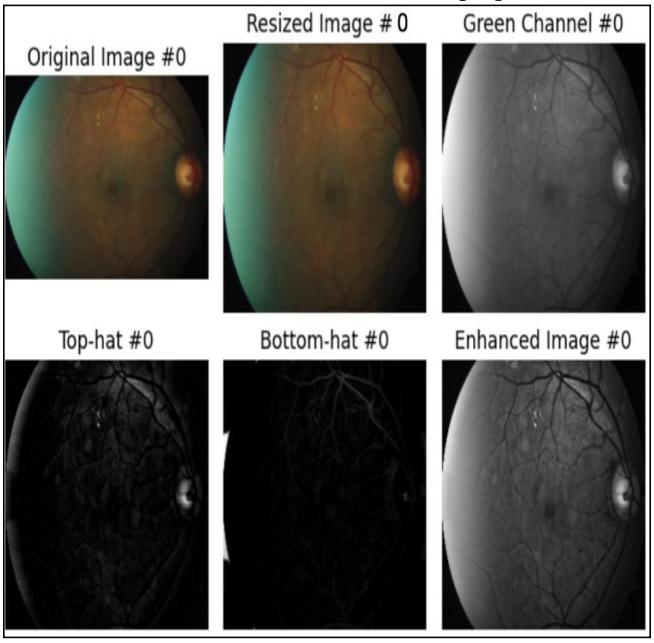


Fig. 1. Preprocessing workflow for diabetic retinopathy analysis applied to retinal fundus images. Row 1 represents: the original image, resized image, green channel extraction, row 2 represents: top-hat filtering, bottom-hat filtering, and the final contrast-enhanced image.

3) Top-hat and bottom-hat Transformations: These image morphisms are employed to improve an image's visual quality; they are particularly useful in fields such as medical imaging. The top-hat modification [36] is a technique that reduces the picture opening from the initial image, highlighting small, bright areas on the image, known as exudates, as depicted in the figure. This method supports the idea that it may be able to identify the minuscule traits that are probably very early indicators of DR. This is aided by the bottom-hat transformation [37], which subtracts the original image from the image's closure to locate dark patches such as hemorrhages. As demonstrated in Fig. 1, these modifications, which assist the model in focusing on minor but crucial aspects, might be highly advantageous for diseases and the lesions associated with them.

C. Deep Feature Extraction

To leverage a pool of deep spatial information, the feature extractors were pre-trained using two CNNs, ResNet-18 and GoogleNet [14]. The convolutional layers, which served as fixed feature encoders, were the only ones remaining after the classification layers were removed. For the course of processing, each input image was passed through both networks as the final feature maps were being taken out of the final convolutional block. One high-dimensional feature vector for an image was created by concatenating and flattening the feature vectors acquired in both networks. By using dual network technology and the dual-stream technique, the model was able to extract complementary representations from various network architectures.

D. Z-Score Normalization

Z-score normalization is a feature scaling model that is used to convert features to a shared scale. Data is processed to Z-score normalization, which involves the standardization of each feature. The normalized value is determined by the following Eq. (1):

$$z = \frac{x-\mu}{\sigma} \quad (1)$$

where, x represents the original feature, μ represents the feature's mean, and σ represents the standard deviation. The distribution will then have a mean of 0 and a standard deviation of 1. Z-score normalization is particularly beneficial when the features are on different scales or have different units, as it standardizes them, thereby improving the performance and convergence of numerous ML algorithms, including SVM, KNN, and gradient-based models [38].

E. Classification Algorithms

The normalized features were used to train and evaluate a set of traditional machine learning classifiers [15], viz., AdaBoost, Decision Tree (DT), Gradient Boosting (GBM), Random Forest (RF), Extreme Gradient Boosting (XGBoost), K-Nearest Neighbors (KNN), Gaussian Naive Bayes (GNB), Light Gradient Boosting Machine (LightGBM), and Support Vector Machine (SVM) with Linear and RBF kernels.

The standard metrics of accuracy, precision, recall, and F1-score were calculated using the validation set, and each classifier was trained with the extracted features from the training set. To resolve the imbalance problem, all metrics were class-weighted [15]. Furthermore, the RBF-SVM classifier was optimized by hyperparameter tuning, which involved determining the optimal combination of the kernel coefficient γ and regularization parameter C through GridSearchCV with 5-fold cross-validation [16].

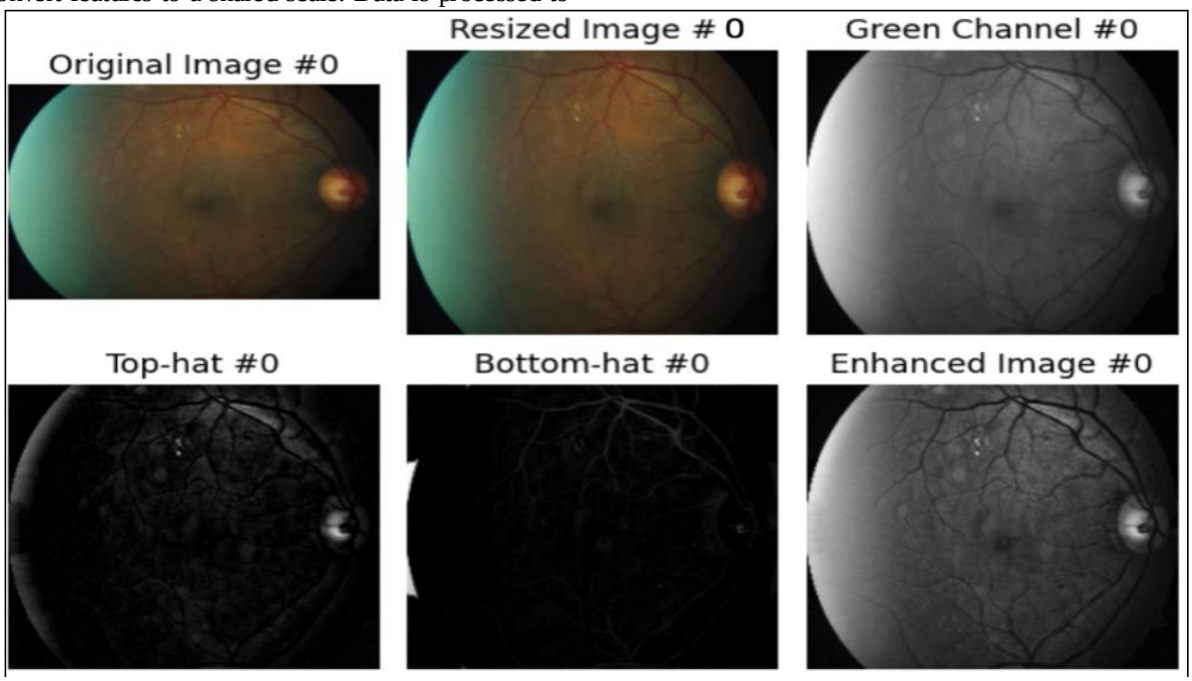


Fig. 2. Proposed algorithm.

F. Proposed Algorithm

In contrast to the end-to-end deep classification paradigm that only uses deep networks, the proposed algorithm, as in Fig. 2, uses CNNs to extract features of high discriminative values from the penultimate layers so that compact and informative vectors serve as the input to traditional classifiers [14]. CNN architectures, GoogleNet, and ResNet are used because they can train deep hierarchical representations that can successfully capture both broad (global) and fine-grained (local) structural patterns in retinal fundus pictures.

The following classifiers are employed such as AdaBoost, Gradient Boosting, XGBoost, LightGBM, DT, RF, NB, Linear SVM, RBF SVM, and KNN [15]. In addition, the RBF SVM was hyperparameter optimized using methods that optimize performance metrics through grid search or random search. Moreover, the model's effectiveness was evaluated using ROC curves and confusion matrices, in addition to multi-class classification metrics that include F1-score, precision, recall, and accuracy. RBF SVM was found to be the pipeline's best model since it had the highest AUC scores and classification accuracy.

The code is available at https://github.com/amancheema2k12/https-drive.google.com-file-d-1Pbmoh5P1r4RNFlIf6RstIPLE9BYImlOc-view-usp-sharing.

IV. Results

A. Overall Performance

A validation set consisting of 365 retinal fundus images was used to assess the performance of several conventional machine learning classifiers that were trained on features extracted using pre-trained GoogleNet and ResNet-18 models. Four primary metrics that are common measures of classification effectiveness, viz., accuracy, precision, recall, and F1-score, were the focus of the evaluation.

Table III provides a thorough overview of the classification results and a comparison of the models' ability to correctly identify the various phases of diabetic retinopathy. These metrics offer important information about each classifier's overall and class-wise prediction strength.

TABLE III. Performance Metrics Evaluation of Each Classifier

Classifier	Accuracy	Precision	Recall	F1-Score
KNN	0.75	0.76	0.75	0.74
Random Forest	0.78	0.79	0.78	0.77
Decision Tree	0.70	0.71	0.70	0.69
Naive Bayes	0.68	0.69	0.68	0.67
Linear SVM	0.76	0.76	0.76	0.75
AdaBoost	0.77	0.78	0.77	0.76
Gradient Boosting	0.80	0.81	0.80	0.79
XGBoost	0.81	0.82	0.81	0.80
LightGBM	0.82	0.83	0.82	0.81
RBF SVM	0.85	0.85	0.85	0.85

B. Analysis and Observations

According to Fig. 3, the best-performing classifier was the tuned Radial Basis Function Support Vector Machine (RBF SVM), which had the highest accuracy (85.48%), precision (85.05%), and F1-score (84.93%). Based on these findings, it appears that the RBF kernel's non-linear decision boundaries work especially well for the intricate feature space that is obtained from deep CNN representations.

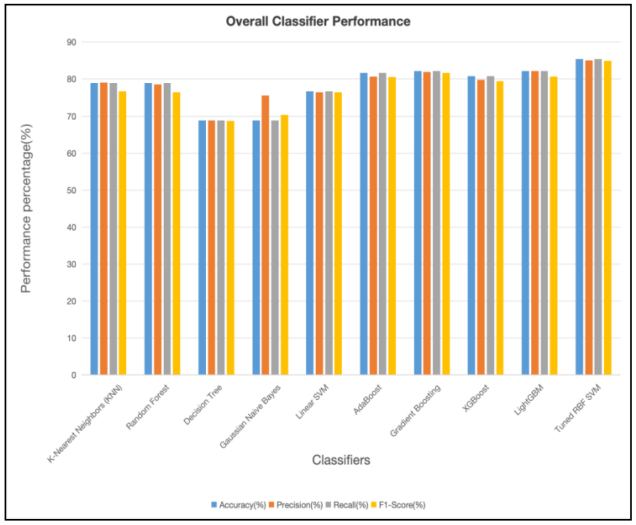


Fig. 3. Performance metrics for the various classifiers.

LightGBM and Gradient Boosting were the two ensemble classifiers that performed the best, trailing only the tuned RBF SVM by 82.19% and 83.01%, respectively. Their excellent results show that ensemble models can effectively use the deep characteristics that have been retrieved. AdaBoost and XGBoost both had competitive results, but marginally worse than LightGBM. Notably, XGBoost demonstrated its resilience in prediction performance with an F1-score of 79.48% and a precision of 79.86%.

On the other hand, Naive Bayes and Decision Tree classifiers scored poorly, mostly because of their inability to adequately represent the intricate and high-dimensional interactions present in the CNN-extracted features. Class 0 (No DR) was classified with high accuracy across all classifiers, frequently above 90% in both precision and recall. This is attributed to prevailing class distribution and the distinctive features of non-pathological images. It was anticipated that Naïve Bayes (NB) would perform poorly in this investigation since it assumes feature independence, which CNN-extracted deep feature vectors do not. Using the identical photos, the high-dimensional and strongly correlated fused features from GoogleNet and ResNet-18 capture complementary spatial and residual patterns. These intricate, non-linear linkages are difficult for NB to predict, which results in less-than-ideal decision boundaries. Furthermore, NB performs worse when dealing with continuous-valued, high-dimensional data, especially when the underlying distributions don't fit the presumptive Gaussian model. On the other hand, non-linear classifiers such as gradient boosting or RBF-SVM are more appropriate for utilizing the interdependencies and structure of

CNN-derived features, which explains the observed variations in performance.

Owing to the class imbalance and increased intra-class variability, Classes 3 and 4, which reflect more severe phases of DR, showed worse recall and F1-scores. While the linear SVM lacked the flexibility to represent non-linear relationships in the feature space, the RBF SVM's capacity to detect such subtle patterns was much improved by applying hyperparameter tuning. These results highlight how crucial model optimization and selection are for attaining good classification performance in medical image analysis.

C. Confusion Matrix Analysis

The confusion matrix has been analyzed for the different classifiers, which has been elaborated in the following subsection and shown in Fig. 4.

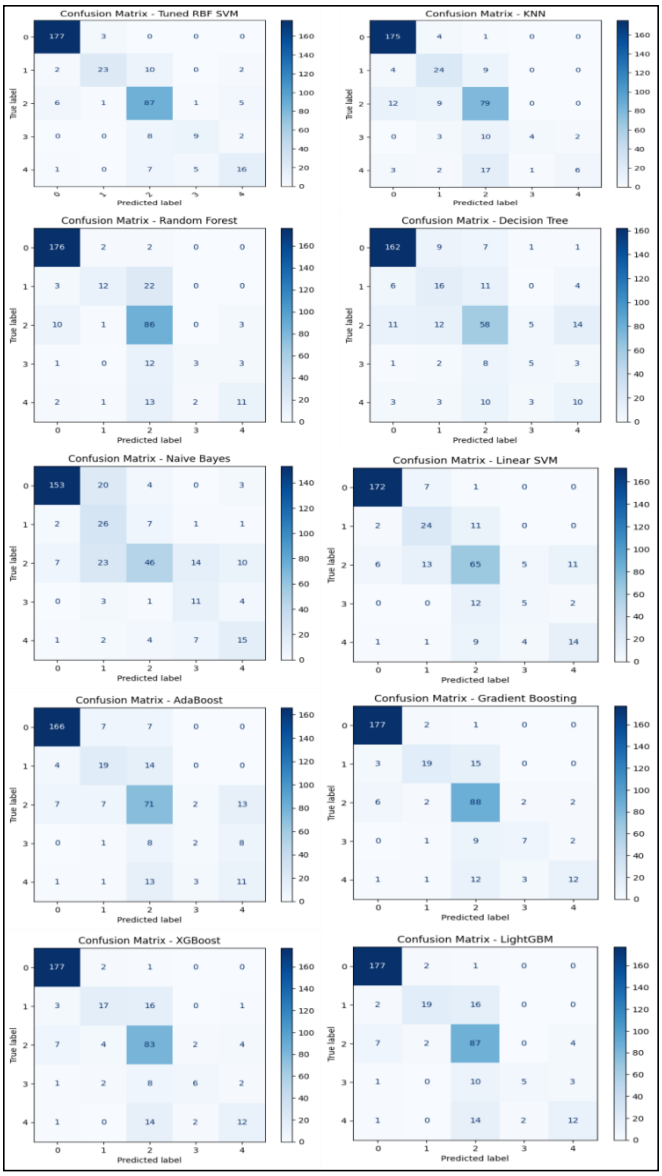


Fig. 4. Confusion matrix for different classifiers employed.

1) Tuned RBF-SVM: The confusion matrix of the best model (Tuned RBF SVM) displayed a remarkable classification for Class 0 (No DR), as 177 out of 180 were correctly classified, as shown in Fig. 4. Class 2 (Moderate DR) also performed well, having 87 correct predictions. However, some confusion appeared between Classes 1 to 2 and 3 to 4 due to the overlapping features at the early stages of DR.

2) Light GBM and XGBoost: Both Light GBM and XGBoost models showed parallel performance, particularly regarding Classes 0 and 2. The misclassifications were mainly in Classes 1 and 4. LightGBM made 177 correct predictions in Class 0 and 87 in Class 2, but struggled with Class 4 (only 12 correct out of 29).

3) Gradient boosting: Gradient boosting found a good mixture, with 88 Class 2 samples per prediction correct, along with 177 Class 0. Minor confusion took place between Classes 1 and 2, as well as underperformance recorded for Class 4.

4) AdaBoost: AdaBoost recorded average showings, with the predominant confusion being in Class 4 (13 calculated as Class 2). There were also several instances of the miscalculated Class 2, which was in Class 4.

5) Linear SVM: Linear SVM observed decreased success due to being unable to differentiate between moderate to severe DR stages (Classes 2-4). Class 2 had a miserable total of 65 correct predictions, with a notable number being misclassified as Class 4.

6) Naïve Bayes and Decision Tree: These two were the worst hit so far. The Decision Tree managed only to classify 58 instances in Class 2, with the other classes being highly confused. Naive Bayes classified better in Classes 0 and 1, but gave up drastically for Classes 2-4.

7) kNN and random forest: kNN and Random Forest exhibited similar performance. kNN had its ups and downs in Classes 2 and 4, while Random Forest did a bit better, with Class 2 having 86 and Class 0 having 176 being correctly classified.

D. ROC Curve Analysis

1) To further evaluate each classifier's discriminative performance, Receiver Operating Characteristic (ROC) curves were used, as seen in Fig. 5. From Class 0 (No DR) to Class 4 (Proliferative DR), these graphs evaluate the true positive rate (sensitivity) with the false positive rate for each of the five diabetic retinopathy (DR) classes. With Area Under the Curve (AUC) values of 0.99, 0.87, 0.92, 0.91, and 0.92 for Classes 0 through 4, respectively, the tweaked RBF SVM outperformed all other models in the evaluation. These findings highlight the model's strong multi-class separability and steady dependability at different DR severity levels.

2) Following closely behind, XGBoost and LightGBM likewise obtained AUC ratings ≥ 0.92 in every class, demonstrating their efficacy, especially in differentiating more complex DR stages, where other models generally falter. Particularly for Classes 3 and 4, which are frequently

underrepresented and more difficult to categorize, these ensemble-based approaches demonstrated proficiency in capturing intricate decision limits.

3) Additionally, AUC values above 0.90 were maintained by Random Forest, AdaBoost, and Gradient Boosting, indicating strong learning from the retrieved features and good generalization. However, the AUC scores of KNN and Decision Tree classifiers were much lower, especially for the higher-grade DR classes, suggesting that they had a limited ability to model the subtle variations required for accurate classification. While achieving moderate performance, Naive Bayes and Linear SVM were unable to effectively distinguish the more severe DR categories, falling short of the top-tier classifiers.

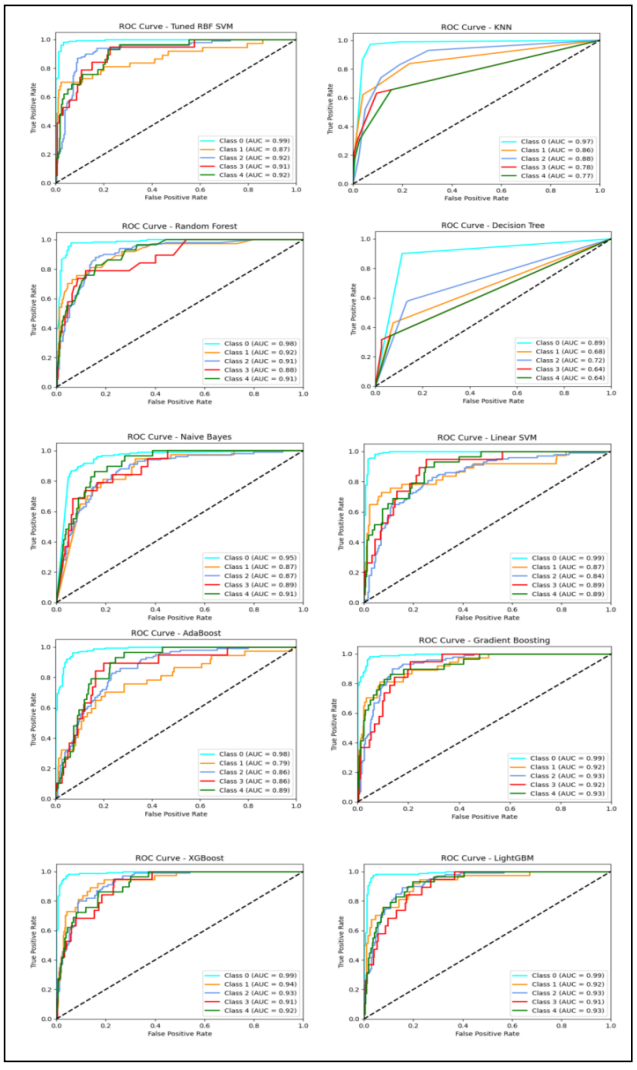


Fig. 5. ROC curves for the classifiers employed across different sub-classes.

4) The tuned RBF SVM, XGBoost, and LightGBM are the most efficient and dependable models for real-world DR diagnostic applications, where precise classification across all disease stages is essential for prompt intervention and treatment planning, according to the ROC analysis.

E. Paired t-tests

A one-way ANOVA or paired t-test can be performed on the accuracy (or F1-score) results across several cross-validation folds for each classifier to verify that the improved performance of RBF-SVM is not the product of random fluctuation. The paired *t*-tests across 5-fold cross-validations are shown in Table IV.

TABLE IV. PAIRED *T*-TESTS

Classifier	Accuracy	p-value vs RBF-SVM
RBF-SVM	85.48	--
XGBoost	83.10	0.032
Random Forest	81.75	0.018
Gradient Boosting	82.20	0.0027
Logistic Regression	78.90	0.005
Decision Tree	74.35	0.001
Naive Bayes	69.80	<0.001

V. DISCUSSION

1) Effectiveness of the hybrid CNN-ML framework: The suggested hybrid system effectively blends conventional machine learning classifiers with deep feature extraction utilizing GoogleNet and ResNet-18. When compared to standalone end-to-end deep learning models, our modular design improves diagnostic performance. A more discriminative feature space for classification is produced by the dual-CNN setup's efficient acquisition of complementary feature representations.

2) Performance of classifiers: With excellent precision, recall, and F1-score values, the RBF-SVM outperformed the other ten classifiers in terms of accuracy, achieving 85.48 per cent. The findings show that non-linear classifiers can even beat sophisticated ensemble techniques like XGBoost and LightGBM in terms of using deeply abstracted information. On the other hand, less complex models such as Decision Tree and Naïve Bayes performed poorly, demonstrating their inability to handle non-linear and high-dimensional feature fields. RBF-SVM performs better than other models because it is especially well-suited to handle the structured, high-dimensional, and non-linear data that CNNs extract. It also performs better in generalization and minority class separation.

3) Analysis of class imbalance: Confusion matrix and ROC studies showed that the hybrid RBF-SVM was better at separating minority classes, especially Proliferative DR (Class 4), than the majority models, which did well at recognizing the majority class (No DR). This implies that the hybrid approach improves sensitivity to underrepresented but clinically important categories.

4) Comparison of existing literature: The suggested model's performance was contrasted with earlier hybrid methodologies. On the APTOS 2019 dataset, for example, Mohanty et al. (2023) used a VGG16 + XGBoost pipeline and reported 79.50% accuracy. In contrast, our hybrid CNN–ML

approach provided more resilience and diversity of features in addition to increased accuracy. The suggested hybrid pipeline might initially see a performance drop if used on bigger and more varied datasets like EyePAC because of domain shift brought on by variations in image quality, acquisition methods, and class distributions in comparison to APTOS 2019. This restriction can be lessened, though, by fine-tuning the CNN feature extractors on EyePACS, re-optimizing the RBF-SVM classifier on the new feature space, and adjusting preprocessing procedures, viz., illumination normalization and top-hat filtering.

5) Clinical and practical implications: The framework is ideal for implementation in resource-limited settings, where end-to-end deep models might be computationally prohibitive due to its modular design. The suggested process enables the reuse of derived embeddings for related tasks like disease progression analysis and severity prediction by separating feature extraction from classification. Furthermore, interpretability is enhanced because classical classifiers offer decision bounds and feature importance ratings, which promote clinician confidence in AI-assisted systems.

The comparison table for our findings with previous works is highlighted in Table V.

TABLE V. COMPARATIVE ANALYSIS OF PREVIOUS WORK WITH CURRENT WORK

Author (s) and Year	Methodology	Dataset	Accuracy	Key Contributions/Limitations
Pratt et al. (2016) [8]	CNN trained end-to-end	Kaggle DR	75%	First CNN-based DR detection; limited generalizability
Gulshan et al. (2016) [40]	INCEPTION V3	EyePACS	82%	Large-scale DR screening faced a class imbalance
Voets et al. (2019) [41]	RESET-50	Messidor, EyePACS	79.5%	Showed transfer learning potential; imbalance issues
Islam et al. (2022) [33]	CNN+SVM	Kaggle DR	81.7%	Demonstrated hybrid approach; moderate performance
Mohanty et al. (2023) [39]	VGG-16+Xgboost	APTOS 2019	79.5%	Hybrid pipeline; less effective than CNN+SVM fusion
Our Work	Dual-CNN (GoogleNet + ResNet-18) + RBF-SVM	APTOS 2019	85.48%	Higher accuracy, better detection of minority classes, improved interpretability, and resource-efficient design

VI. CONCLUSION

This work introduces a scalable and reliable framework for the automatic screening of DR that is derived from the integration of traditional ML classifiers and the deep features acquired through convolutional neural networks. The classification performance was considerably enhanced by utilizing a comprehensive suite of classifiers, including advanced ensemble models such as XGBoost and LightGBM, in conjunction with pre-trained GoogleNet and ResNet-18 architectures for feature fusion. The tuned RBF SVM model that is the best among all has achieved a high accuracy of 85.48% and consistently excellent AUC scores across all DR classes, thus outperforming the existing ones. Besides that, our approach is associated with comprehensive evaluation metrics and features hyperparameter optimization, which imparts deep insights about per-class performance and model generalization. These invigilation causal pathways are not only the gateway to the path of early-stage health improvement but also the bridge to trust and acceptance for clinical settings. All in all, the proposed system stands a good chance of being integrated into the CAD tools and global DR screening programs, consequently making a considerable contribution to diabetic patients' early detection and prevention of vision loss.

VII. LIMITATIONS AND FUTURE WORK

This study has some limitations despite its encouraging findings. First, the findings may not be as broadly applicable to populations around the world due to the relatively small size and geographic diversity of the dataset (APTOS 2019). Secondly, class imbalance persisted despite the hybrid framework's performance improvement, especially for advanced stages of DR. Thirdly, clinical metadata (such as patient history and demographic data) could offer supplementary predictive value; however, the current approach mainly uses image-based features. Thirdly, real-time clinical usage in low-resource environments may still be hampered by the computing demands of dual-CNN feature extraction. Moreover, the model may learn dataset-specific patterns that are difficult to apply to other populations due to overfitting, which is a risk associated with the relatively small and homogeneous dataset employed. Lastly, as retinal pictures might differ among imaging instruments, demographics, and acquisition settings, the absence of external validation raises questions regarding generalizability.

Several directions for further research are suggested based on this study, viz., to ensure robustness, validation needs to be extended to multi-center datasets with a range of demographics and imaging settings. Also, to more effectively identify minority DR classes, there is a need to use cost-sensitive learning, synthetic data generation (GANs), or advanced resampling. Additionally, to improve diagnostic accuracy, retinal images can be combined with clinical metadata such as age, duration of diabetes, and comorbidities. Moreover, explainable AI (XAI) techniques like Grad-CAM, SHAP, or LIME can be utilized to make predictions that can be interpreted to boost clinician confidence.

REFERENCES

[1] M. Kropp et al., "Diabetic retinopathy as the leading cause of blindness and early predictor of cascading complications—risks and mitigation," EPMA J., vol. 14, no. 1, pp. 21–42, 2023, doi: 10.1007/s13167-023-00314-8.

[2] V. J. Vieira-Potter, D. Karamichos, and D. J. Lee, "Ocular complications of diabetes and therapeutic approaches," Biomed. Res. Int., vol. 2016, p. 3801570, 2016, doi: 10.1155/2016/3801570.

[3] T. Y. Wong and T. E. Tan, "The diabetic retinopathy 'pandemic' and evolving global strategies: The 2023 Friedenwald Lecture," Invest. Ophthalmol. Vis. Sci., vol. 64, no. 15, p. 47, 2023, doi: 10.1167/iovs 64.15.47.

[4] H. Li et al., "Current research and future strategies for the management of vision-threatening diabetic retinopathy," Asia-Pac. J. Ophthalmol., vol. 13, no. 5, p. 100109, 2024, doi: 10.1016/j.apjo.2024.100109.

[5] I. P. Chatziralli, "The value of fundoscopy in general practice," Open Ophthalmol. J., vol. 6, pp. 4–5, 2012.

[6] A. Grzybowski et al., "Retina fundus photograph-based artificial intelligence algorithms in medicine: A systematic review," Ophthalmol. Ther., vol. 13, pp. 2125–2149, 2024, doi: 10.1007/s40123-024-00981-4.

[7] J. Benbassat and B. C. Polak, "Reliability of screening methods for diabetic retinopathy," Diabet. Med., vol. 26, no. 8, pp. 783–790, 2009.

[8] H. Pratt, F. Coenen, D. M. Broadbent, S. P. Harding, and C. Y. Zheng, "Convolutional neural networks for diabetic retinopathy," in Proc. Int. Conf. Med. Imaging Understand. Anal. (MIUA 2016), Loughborough, U.K., Jul. 6–8, 2016, Procedia Comput. Sci., vol. 90, pp. 200–205.

[9] W. Al Ayoubi, W. M. Shalash, and M. F. Abulkhair, "Diabetic retinopathy detection through deep learning techniques: A review," Inf. Med. Unlocked, vol. 20, pp. 1–11, 2020

[10] I. H. Sarker, "Deep learning: A comprehensive overview on techniques, taxonomy, applications and research directions," SN Comput. Sci., vol. 2, p. 420, 2021, doi: 10.1007/s42979-021-00815-1.

[11] X. Zhaoet al., "A review of convolutional neural networks in computer vision," Artif. Intell. Rev., vol. 57, p. 99, 2024, doi: 10.1007/s10462-024-10721-6.

[12] B. Xu and G. Yang, "Interpretability research of deep learning: A literature survey," Inf. Fusion, vol. 115, p. 102721, 2025, doi: 10.1016/j.inffus.2024.102721.

[13] A. E. Ilesanmi, T. Ilesanmi, and G. A. Gbotoso, "A systematic review of retinal fundus image segmentation and classification methods using convolutional neural networks," Healthc. Anal., vol. 4, p. 100261, 2023, doi: 10.1016/j.health.2023.100261.

[14] V. Singh et al., "A hybrid deep learning model for enhanced structural damage detection: Integrating ResNet50, GoogLeNet, and attention mechanisms," Sensors, vol. 24, no. 22, p. 7249, 2024, doi: 10.3390/s24227249.

[15] K. Mani and H. Rajaguru, "A framework for performance enhancement of classifiers in detection of prostate cancer from microarray gene," Heliyon, vol. 10, no. 9, p. e29630, 2024, doi: 10.1016/j.heliyon. 202.e29630.

[16] U. P. S. Parmar et al., "Artificial intelligence (AI) for early diagnosis of retinal diseases," Medicina (Kaunas), vol. 60, no. 4, p. 527, 2024, doi: 10.3390/medicina60040527.

[17] X. Cao, J. Lin, X. Gao, and Z. Li, "Integrating convolution and transformer for enhanced diabetic retinopathy detection," Int. J. Bio-Inspired Comput., vol. 23, no. 4, pp. 225–235, 2024.

[18] X. Luo et al. "A deep convolutional neural network for diabetic retinopathy detection via mining local and long-range dependence," CAAI Trans. Intell. Technol., vol. 9, no. 1, pp. 153–166, 2024.

[19] A. Khaparde, S. Chapadgaonkar, M. Kowdiki, and V. Deshmukh, "An attention-based Swin U-Net segmentation and hybrid deep learning-based diabetic retinopathy classification framework using fundus images," Sens. Imaging, vol. 24, no. 1, p. 20, 2023.

[20] X. Li et al., "Lesion-attention pyramid network for diabetic retinopathy grading," Artif. Intell. Med., vol. 126, p. 102259, 2022.

[21] R. Bala, A. Sharma, and N. Goel, "Comparative analysis of diabetic retinopathy classification approaches using machine learning and deep learning techniques," Arch. Comput. Methods Eng., vol. 31, no. 2, pp. 919–955, 2024.

[22] P. L. Akella and R. Kumar, "An advanced deep learning method to detect and classify diabetic retinopathy based on color fundus images," Graefe's Arch. Clin. Exp. Ophthalmol., vol. 262, no. 1, pp. 231–247, 2024.

[23] P. Saranya, R. Pranati, and S. S. Patro, "Detection and classification of red lesions from retinal images for diabetic retinopathy detection using deep learning models," Multimed. Tools Appl., vol. 82, no. 25, pp. 39327–39347, 2023.

[24] D. J. B. Saini et al., "Convolution neural network model for predicting various lesion-based diseases in diabetic macula edema in optical coherence tomography images," Biomed. Signal Process. Control, vol. 86, p. 105180, 2023.

[25] S. Agarwal and A. Bhat, "A survey on recent developments in diabetic retinopathy detection through integration of deep learning," Multimed. Tools Appl., vol. 82, no. 11, pp. 17321–17351, 2023.

[26] S. Dubey and M. Dixit, "Recent developments on computer-aided systems for diagnosis of diabetic retinopathy: A review," Multimed. Tools Appl., vol. 82, no. 10, pp. 14471–14525, 2023.

[27] V. Vijayan and A. Salim, "Survey on deep learning-based automated systems for the detection and grading of diabetic retinopathy using retinal fundus images," in Proc. 2023 Int. Conf. Power, Instrum., Control Comput. (PICC), IEEE, 2023, pp. 1–6.

[28] M. Z. Atwany, A. H. Sahyoun, and M. Yaqub, "Deep learning techniques for diabetic retinopathy classification: A survey," IEEE Access, vol. 10, pp. 28642–28655, 2022.

[29] M. S. Farooq et al., "Untangling computer-aided diagnostic system for screening diabetic retinopathy based on deep learning techniques," Sensors, vol. 22, no. 5, p. 1803, 2022.

[30] P. Bidwai, S. Gite, K. Pahuja, and K. Kotecha, "A systematic literature review on diabetic retinopathy using an artificial intelligence approach," Big Data Cogn. Comput., vol. 6, no. 4, p. 152, 2022.

[31] B. Lalithadevi and S. Krishnaveni, "Detection of diabetic retinopathy and related retinal disorders using fundus images based on deep learning and image processing techniques: A comprehensive review," Concurrency Comput.: Pract. Exper., vol. 34, no. 19, p. e7032, 2022.

[32] A. M. Dayana and W. S. Emmanuel, "A comprehensive review of diabetic retinopathy detection and grading based on deep learning and metaheuristic optimization techniques," Arch. Comput. Methods Eng., vol. 30, no. 7, pp. 4565–4599, 2023.

[33] M. R. Islam et al., "Applying supervised contrastive learning for the detection of diabetic retinopathy and its severity levels from fundus images," Comput. Biol. Med., vol. 146, p. 105602, 2022, doi: 10.1016/j.compbiomed.2022.105602.

[34] M. Li, Y. Jiang, Y. Zhang, and H. Zhu, "Medical image analysis using deep learning algorithms," Front. Public Health, vol. 11, p. 1273253, 2023, doi: 10.3389/fpubh.2023.1273253.

[35] S. Biswas et al., "Which color channel is better for diagnosing retinal diseases automatically in color fundus photographs?" Life (Basel), vol. 12, no. 7, p. 973, 2022, doi: 10.3390/life12070973.

[36] X. Bai, "Image analysis through feature extraction by using top-hat transform-based morphological contrast operator," Appl. Opt., vol. 52, no. 16, pp. 3777–3789, 2013, doi: 10.1364/AO.52.003777.

[37] A. Malhi, R. Grewal, and H. S. Pannu, "Detection and diabetic retinopathy grading using digital retinal images," Int. J. Intell. Robot. Appl., vol. 7, pp. 426–458, 2023, doi: 10.1007/s41315-022-00269-5.

[38] Y. Ban, D. Zhang, Q. He, and Q. Shen, "APSO-CNN-SE: An adaptive convolutional neural network approach for IoT intrusion detection," Comput. Mater. Continua, vol. 81, no. 1, pp. 567–601, 2024, doi: 10.32604/cmc.2024.055007.

[39] C. Mohanty et al., "Using deep learning architectures for detection and classification of diabetic retinopathy," Sensors (Basel), vol. 23, no. 12, p. 5726, 2023, doi: 10.3390/s23125726.

[40] V. Gulshan et al., "Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs," JAMA, vol. 316, no. 22, pp. 2402–2410, 2016, doi: 10.1001/jama.2016.17216.

[41] M. Voets, K. Møllersen, and L. A. Bongo, "Reproduction study using public data of: Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs," PLOS ONE, vol. 14, no. 6, p. e0217541, 2019, doi: 10.1371/journal.pone.0217541.

Forecasting Currency Exchange Direction with an Advanced Immune-Inspired Model

EL BADAOUI Mohamed[1], RAOUYANE Brahim[2], EL MOUMEN Samira[3], BELLAFKIH Mostafa[4]
Data Engineering and Intelligent Systems Laboratory (LIDSI), Faculty of Sciences, Ain Chock,
HASSAN II University, Casablanca, Morocco[1, 2]
Computer and System Laboratory (LIS), N&IDP Faculty of Sciences, Ain Chock, HASSAN II University, Casablanca, Morocco[3]
Department of Mathematics and Computer Science, RAISS Laboratory,
National Institute of Posts and Telecommunications (INPT), RABAT, Morocco[4]

***Abstract*—Accurately forecasting currency exchange rates is a persistent and significant challenge in computational finance. This study addresses the challenge by introducing an advanced model based on the Artificial Immune Recognition System (AIRS), an algorithm inspired by the adaptive learning of biological immune systems, to predict the directional movement of the EUR/USD pair. While conventional machine learning models are widely used, immune-inspired approaches have been largely unexplored in this domain. Using historical data from May 2002 to July 2024, the proposed model was rigorously optimized through time-series cross-validation and an Evolutionary Algorithm search. On the out-of-sample test set, the optimized model demonstrates strong predictive power, achieving an F1-Score of 0.66 and an ROC AUC of 0.74, results that are competitive with standard machine learning benchmarks. These findings validate AIRS as a robust and scientifically defensible tool for financial forecasting, offering a viable alternative to conventional methods in a highly volatile market.**

Keywords*—Artificial immune recognition system; financial market prediction; machine learning; predictive analytics; time series forecasting

I. Introduction

Predicting financial markets is a notoriously difficult task. The underlying data is noisy, patterns are often fleeting, and traditional statistical models frequently fall short of providing reliable forecasts. Consequently, this has spurred significant research into machine learning approaches.

While a significant body of research has focused on popular models such as neural networks [1] and random forests [2], a fascinating family of algorithms—Artificial Immune Systems (AIS)—has been largely overlooked, especially for the Forex market. This represents a missed opportunity, as the adaptive, self-learning nature of these biological systems seems perfectly suited for the constantly changing market environment.

The objective of this study is therefore not merely to apply another algorithm. Instead, the aim is to rigorously test whether AIRS, a well-known immune-inspired model, can perform competitively in the EUR/USD prediction space. To achieve this, a simple application is insufficient. This work focuses on implementing a robust walk-forward [3] validation method, which is critical for time-series data and involves considerable effort in hyperparameter tuning. The central research question is whether the theoretical promise of AIRS can be translated into a practical, demonstrable edge in financial forecasting.

A. Research Gap and Motivation

A review of the literature reveals a clear disconnect. On one hand, the Artificial Immune Recognition System (AIRS) has been successfully applied in complex fields where it is praised for its ability to learn from noisy and dynamic data. On the other hand, the world of financial prediction is saturated with a wide range of machine learning models [4], yet immune-inspired approaches are conspicuously absent. This gap was the primary motivation for this work.

The core principles of a biological immune system appear tailor-made for the challenges of financial markets:

- Constantly Changing Markets: A model that is effective today may fail tomorrow. The adaptive nature of AIRS, which constantly refines its "memory cells", is designed precisely for such dynamic environments.
- Exploration vs. Exploitation Trade-off: Successful trading requires a balance between exploiting known, profitable patterns and exploring for new ones. The way AIRS manages its population of "antibodies" through cloning (exploitation) and mutation (exploration) directly mirrors this fundamental trade-off.

Given this strong theoretical fit, the lack of rigorous studies applying AIRS to currency markets constitutes a significant oversight. This study was therefore designed to answer a simple question: Can a well-tuned AIRS model provide a real, practical edge in predicting the direction of the EUR/USD pair?

B. Main Contributions and Novelty

The primary contribution of this work is the first rigorous application and validation of the AIRS algorithm for EUR/USD exchange rate prediction. The study moves beyond a simple proof-of-concept by introducing a robust framework that integrates time-series-aware validation (walk-forward cross-validation) and advanced hyperparameter optimization (Evolutionary Algorithm Search CV), addressing a common weakness in prior financial forecasting studies. Furthermore, a practical implementation is provided that tackles the specific challenges of financial data, from feature engineering with technical indicators to handling temporal dependencies. To

*Corresponding Author.

support transparency and reproducibility, the mathematical underpinnings of the AIRS model are also detailed in this specific context.

C. Paper Organization

The remainder of this study is structured as follows: Section II reviews related work in the field. Section III details the methodology used for the study. Section IV presents the mathematical framework of the AIRS model and the experimental setup. Section V presents the empirical results, which are interpreted and discussed in Section VI. Finally, Section VII concludes the study by summarizing the key findings and suggesting avenues for future research.

II. Related Work

This section provides a comprehensive review of the literature pertinent to financial market forecasting, establishing the context and highlighting the research gap that this study addresses. The review is structured into key areas: the application of conventional machine learning and deep learning models to currency exchange prediction, and the use of bio-inspired algorithms in finance, with a specific focus on Artificial Immune Systems.

A. Machine Learning Approaches for Currency Exchange Forecasting

The prediction of currency exchange rates has long been a focal point of research in computational finance. A wide array of machine learning models has been employed to tackle this challenging task. Early research often focused on traditional models such as Support Vector Machines (SVMs), which have demonstrated effectiveness in classification tasks due to their ability to handle non-linear data [5]. Similarly, ensemble methods like Random Forests have been widely adopted, valued for their robustness and their ability to mitigate overfitting by aggregating the predictions of multiple decision trees [2].

More recently, the field has seen a surge in the application of deep learning models, particularly Long Short-Term Memory (LSTM) networks. Given their inherent capacity to capture long-term dependencies in sequential data, LSTMs have shown remarkable promise. For instance, several studies report high statistical accuracy, often exceeding 90%, when using LSTMs for trend forecasting [1].

However, a critical review of this literature reveals significant limitations. Many works achieving exceptionally high accuracy metrics do so under idealized conditions, failing to account for market frictions such as transaction costs or slippage. Furthermore, as highlighted by Khan et al. [2], high statistical accuracy does not necessarily translate to economic profitability, and a substantial number of studies lack rigorous financial backtesting. This gap between statistical performance and practical utility underscores the need for models that are not only accurate but also validated through methodologically sound frameworks.

B. Bio-Inspired Algorithms and Artificial Immune Systems

Bio-inspired computing, which draws inspiration from natural systems, offers a promising alternative to conventional machine learning models. Algorithms such as Genetic Algorithms and Particle Swarm Optimization have been successfully applied to problems like portfolio optimization and trading rule discovery. Within this domain, Artificial Immune Systems (AIS) stand out due to their adaptive, self-learning nature, which mirrors the human immune system's ability to recognize and respond to novel pathogens [6].

The Artificial Immune Recognition System (AIRS), first introduced by Watkins et al. [7], is a supervised learning algorithm inspired by the principles of clonal selection and affinity maturation. Its foundational design has proven to be a robust classifier, leading to numerous enhancements. For example, variations like AIBARS [8] have improved computational efficiency, while O-AIRS [9] was developed specifically to address the challenge of overfitting in complex datasets.

The versatility of AIRS is evident in its successful application across diverse fields. In finance, it has been adapted for high-stakes problems such as fraud detection [10] and credit rating prediction [11]. Its capacity to handle noisy, high-dimensional data has also been proven in medical diagnostics, where it achieved high accuracy in tasks like hepatitis classification [12].

Despite these successes, a significant research gap persists. While AIS and AIRS in particular have demonstrated their power, their potential remains largely unexplored in the specific context of financial time-series prediction. The dynamic, non-stationary, and noisy nature of the foreign exchange market appears ideally suited to the adaptive capabilities of AIRS. This study aims to bridge this gap by conducting the first rigorous application and validation of an optimized AIRS model for predicting the directional movement of the EUR/USD pair. Building upon preliminary work that applied machine learning to forecast Bitcoin's direction [13], this study extends that research to the unique and complex dynamics of a major currency market.

III. Methodology

This section details the comprehensive framework used to develop, train, and evaluate the proposed forecasting model. It begins by presenting the mathematical formulation of the Artificial Immune Recognition System (AIRS) as adapted for this study. Subsequently, it describes the data collection, feature engineering, hyperparameter optimization, and evaluation protocols.

A. The AIRS Forecasting Model

As illustrated in the conceptual framework (Fig. 1), the AIRS model processes financial data through a well-defined evolutionary workflow to predict the EUR/USD direction. The process begins by initializing a diverse population of candidate solutions, known as antibodies. The model then enters an iterative training loop for each input vector, which consists of four core stages: affinity calculation to measure the fitness of each antibody, followed by cloning and mutation to reproduce and diversify the best candidates. A selection mechanism then retains the most promising antibodies for the next generation. This adaptive cycle, which is detailed in the following subsections, allows the model to effectively learn from complex market dynamics.

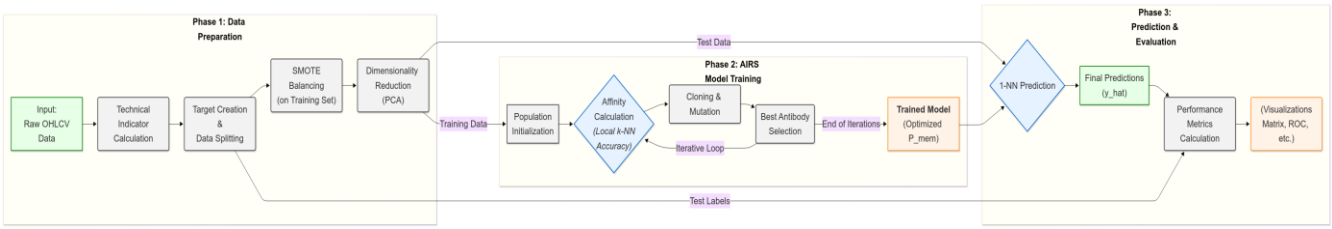


Fig. 1. Conceptual framework of the AIRS model for EUR/USD direction prediction.

1) Problem formulation: Let the dataset be $D = \{\{x_i, y_i\}\}_{i=1}^{n}$, where $x_i \in \mathbb{R}^d$ is a feature vector of financial indicators and $y_i \in \{0,1\}$ is the corresponding target label (1 for 'up', 0 for 'down'). The objective is to learn a classification function $f: \mathbb{R}^d \to \{0,1\}$, modeled by AIRS, that accurately predicts the class y for a new feature vector x.

2) Initialization: The process starts by initializing a population P of M antibodies. Each antibody w_i is a data prototype, defined as a tuple $w_i = (v_i, c_i)$, where:

- $v_i \in \mathbb{R}^d$ is a feature vector representing the antibody's position in the feature space
- $c_i \in \{0,1\}$ is the class label associated with the prototype.

The population is initialized by randomly sampling M data points from the training set D. For each chosen sample (x_j, y_j), an antibody w_i is created such that $v_i = x_j$ and $c_i = y_j$. This strategy ensures that the initial population of prototypes is situated in relevant, data-dense regions of the feature space.

3) Affinity calculation initialization: Affinity measures an antibody's ability to correctly represent its class within its local neighborhood. For a given antibody $w_i = (v_i, c_i)$ and the training set D, we first identify the set $N_k(v_i)$ of the k nearest neighbors to v_i within D. The affinity is then defined as the classification accuracy of w_i on this local subset:

$$Aff(w_i, D, k) = \left(\frac{1}{k}\right) * \Sigma\{x_j \in N_k(v_i)\}\, I(y_j = c_i) \quad (1)$$

where, $(y_j = c_i)$ is the indicator function. This ensures that antibodies representing dense and pure regions of their class are assigned higher affinity.

4) Adaptive cloning and mutation: To explore the solution space and refine the antibody population, a process of reproduction and diversification is employed.

- Cloning: In our implementation, a fixed-rate cloning strategy is employed. For each antibody in the current population, a single clone is generated. This approach ensures a consistent level of exploration by temporarily doubling the population size before the selection phase.
- Mutation: Each resulting clone is then mutated. The mutation rate, β, is a fixed hyperparameter, constant throughout the training process. The mutation is applied by adding a small random vector drawn from a Gaussian distribution to the clone's feature vector:

$$v'_{clone} = v_{original} + \beta \cdot N(0,1). \quad (2)$$

5) Selection and memory cell update: From the pool of mutated clones, the n_best candidates with the highest affinity are selected to form the new population P_new. A key feature of AIRS is the maintenance of a separate memory cell population, P_mem, which stores the best-performing antibodies found during training.

6) Prediction mechanism: For a new input vector x, the prediction is determined using a $1 - Nearest\ Neighbor$ $(1 - NN)$ classification rule based on the final memory cell population, P_mem.

- Calculate the Euclidean distance between the new input vector x and the feature vector v of every memory cell in P_{mem}.
- Identify the single memory cell, $w_closest$, that has the minimum distance to x.
- The final prediction, $\hat{y}$, is simply the class label $c_closest$ associated with this single closest memory cell.

7) Theoretical Properties

a) Complexity analysis: The computational complexity for a dataset of size n, a population of M antibodies, and T epochs is approximately $O(T \cdot n \cdot (M \cdot d + M \log M))$, dominated by the affinity calculation and selection steps within the training loop.

b) Convergence properties: The convergence of AIRS is analyzed as a stochastic process. The elitist selection mechanism ensures that the quality of the best solution found by the algorithm is non-decreasing over time, guaranteeing a progressive convergence towards highly fit regions of the solution space.

8) Numerical Walkthrough

a) Initialization: The process begins by creating an initial population from the training data. For this example, assume a population of two antibodies: w_1, representing class 1 with the feature vector $[0.18, 0.22]$, and w_2, representing class 0 with the feature vector $[0.75, 0.88]$.

b) Affinity calculation: The affinity of each antibody is calculated based on its quality within the training data, where a higher affinity indicates a better representative of its class. For our example, let's assume the calculated affinities are $Aff(w_1) \approx 0.67$ and $Aff(w_2) = 1.0$.

c) Adaptive cloning and mutation: According to the implemented strategy, all antibodies in the population are cloned once and then mutated. Both w_1 $(class\ 1, affinity \approx$

0 and w_2 ($class\ 0, affinity\ =\ 1.0$) will generate one clone each. Their feature vectors are slightly altered by Gaussian noise, resulting in new antibodies such as $w_1'(clone)$ with features $[0.20, 0.25]$ and $w_2'(clone1)$ with features $[0.78, 0.90]$.

d) Selection: The affinities of the original antibodies and the new clones are calculated. The combined pool now consists of four antibodies: w_1 , w_2 , $w_1'(clone)$, and $w_2'(clone)$. From this pool, the best antibodies are selected to form the next generation's population. For instance, if the new affinities are calculated as $Aff(w'\ 1(clone)) \approx 0.75$ and $Aff(w_2'(clone)\ ) \approx 0.98$, and the target population size is 2, the new population would consist of the two antibodies with the highest scores: w_2 and $w_2'(clone)$.

e) Prediction: Once the model is trained, predictions are made using the final memory pool. For a new input vector, such as $x_2 = [0.1, 0.3]$, the algorithm identifies the single closest antibody in the pool ($1 - Nearest\ Neighbor$). If the closest antibody found has features $[0.15, 0.25]$ and class $c = 1$, then the final prediction $\hat{y}$ for the input x_2 will be 1 ($'up'$).

IV. Experimental Setup

This section details the complete methodological framework used to build, train, and evaluate the AIRS forecasting model. It describes the data collection and preparation process, the feature engineering approach, the hyperparameter optimization strategy, and the rigorous validation protocol.

A. Data Collection and Preparation

The historical price data for the EUR/USD currency pair were collected from a recognized trading platform, ensuring reliability and accuracy. The dataset spans from May 1, 2002, to July 31, 2024, and includes daily opening prices, closing prices, highs, lows, and transaction volumes. The raw data were carefully preprocessed to ensure its integrity, which involved handling any missing values through forward-fill imputation and checking for outliers.

To ensure a robust evaluation, the data was chronologically partitioned into three distinct sets: a training set (60% of the data), a validation set (20%), and a final test set (20%). The model was trained and optimized using the training set, evaluated during development on the validation set, and its final generalization performance was assessed on the completely held-out test set.

B. Feature Engineering and Preprocessing

To provide the model with meaningful predictive signals, a comprehensive preprocessing and feature engineering pipeline was implemented.

a) Technical indicators: A suite of widely used technical indicators was calculated to capture market dynamics. These included trend indicators (e.g., Simple Moving Averages over 5, 10, and 20-day periods) [14], momentum indicators (e.g., the 14-day Relative Strength Index (RSI) and the Moving Average Convergence Divergence (MACD) with standard parameters) [15], and volatility indicators (e.g., 20-day Bollinger Bands) [16].

b) Lag features: To capture temporal dependencies, lag features for key variables (e.g., closing price, volume) were created for periods ranging from 1 to 5 previous days.

c) Stationarity: The stationarity of all feature series was checked using the Augmented Dickey-Fuller (ADF) test. Non-stationary series were transformed using first-order differencing to ensure robust modeling.

d) Data transformation: The preprocessing pipeline included several key transformations applied sequentially to the training data:

- Imbalance Handling: The Synthetic Minority Over-sampling Technique (SMOTE) was applied to the training set to correct for class imbalance.
- Scaling: All engineered features were standardized using StandardScaler.
- Dimensionality Reduction: Principal Component Analysis (PCA) was performed to reduce the feature space. The number of principal components was selected to retain 95% of the original variance, thus reducing multicollinearity and model complexity. The same scaling and PCA transformations were subsequently applied to the validation and test sets.

C. Hyperparameter Optimization

The performance of the AIRS model is highly dependent on its hyperparameters. To identify the optimal configuration, this study employed an EvolutionaryAlgorithmSearchCV. This technique intelligently explores the hyperparameter space using principles of evolutionary computation. The search was conducted using a TimeSeriesSplit with 5 folds as the cross-validation strategy to respect the temporal order of the data. The objective was to find the hyperparameter combination that maximized the F1-score. The key tuned hyperparameters and their respective search spaces were:

- n_iterations: Integer values in the range [50, 150].
- population_size: Integer values in the range [100, 300].
- mutation_rate: Float values in the range [0.05, 0.2].
- n_best: Integer values in the range [5, 15].

D. Model Validation and Evaluation Framework

1) Evaluation metrics: A comprehensive set of metrics, detailed in Table I, was employed to assess the model's performance. While accuracy offers a straightforward measure, the F1-score provides a more robust measure of classification quality. From a financial perspective, the Sharpe Ratio is especially significant as it quantifies the risk-adjusted return of an implied trading strategy.

TABLE I. THE FOLLOWING EVALUATION METRICS WERE USED TO ASSESS THE MODEL'S PERFORMANCE

Accuracy	Overall proportion of correct predictions [17].
F1 Score	Harmonic mean of precision and recall, robust to imbalance [18].
Precision and Recall	Measures of the quality and completeness of positive predictions.[19]
ROC AUC	Model's ability to discriminate between the 'up' and 'down' classes [20].
Matthews Correlation Coefficient	Measure of the quality of binary classifications [21].
Cohen's Kappa	Measure of agreement between predicted and actual classes [22].
Sharpe Ratio	Measures the risk-adjusted return of an implied trading strategy [23].

2) Performance diagnostics: Beyond numerical metrics, key visualization techniques were used for in-depth analysis. Confusion Matrices were used to analyze error types (false positives versus false negatives). ROC and Precision-Recall Curves allowed for the evaluation of the model's discrimination ability across different decision thresholds. Finally, Cumulative Gain Curves helped to illustrate the model's efficiency in identifying positive cases, providing insight into its practical value for targeted trading strategies.

V. RESULTS

This section presents the empirical results of the optimized AIRS model for predicting the directional movement of the EUR/USD exchange rate. The overall performance of the model on the out-of-sample test set is first detailed, followed by an in-depth diagnostic analysis and a comparative benchmark against standard machine learning models.

A. Overall Predictive Performance

The performance of the optimized AIRS model on the out-of-sample test set demonstrates its predictive capability. The model achieved an accuracy of 66.46% and a robust F1-score of 0.6583. This result indicates a solid balance between a strong precision of 0.7036 and a recall of 0.6185, suggesting the model is more reliable in its positive predictions than exhaustive in identifying all positive cases. The model's ability to discriminate between upward and downward movements is confirmed by a ROC AUC of 0.7406. The detailed performance metrics for both the validation and test sets are summarized in Table II.

B. In-Depth Diagnostic Analysis

To further understand the model's behavior, its performance diagnostics were analyzed, as shown in Fig. 2. The confusion matrix for the test set recorded 389 true negatives and 368 true positives, against 155 false positives and 227 false negatives. This reveals that the model is slightly more prone to missing a potential gain (a false negative) than to incorrectly predicting one (a false positive), a conservative characteristic that can be desirable in risk-averse strategies.

TABLE II. PERFORMANCE METRICS OF THE AIRS MODEL ON EUR/USD DIRECTION PREDICTION

Metric	Validation Set	Test Set
Accuracy	0.6831	0.6646
F1 Score	0.6727	0.6583
Precision	0.7040	0.7036
Recall	0.6441	0.6185
ROC AUC	0.7503	0.7406
MCC	0.3680	0.3343
Kappa	0.3667	0.3317
Sharpe Ratio	0.0766	0.1098

The diagnostic plots confirm the model's robustness. The ROC curve (AUC = 0.74) and the Precision-Recall curve (Average Precision = 0.76) both indicate strong class separation. The cumulative gain curve is particularly insightful, demonstrating that, by targeting the top 40% of the model's most confident predictions, an investor could capture approximately 70% of all actual positive market movements, highlighting the model's practical efficiency.

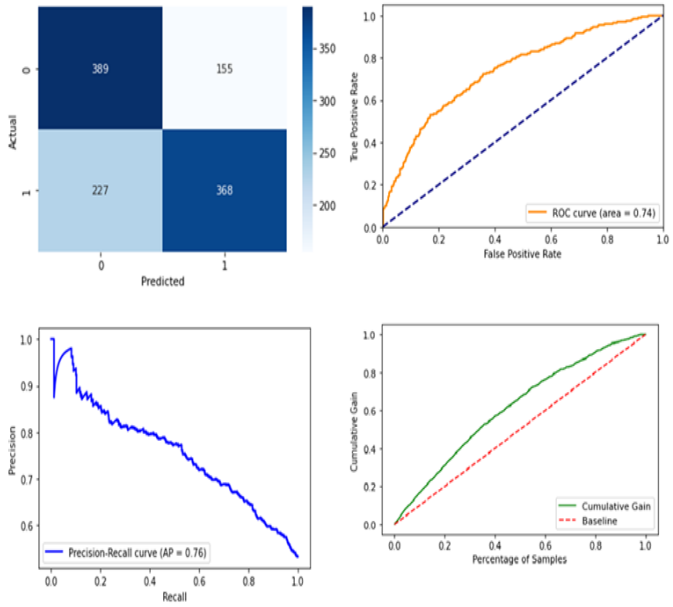


Fig. 2. Performance evaluation metrics of the AIRS model on the test set.

C. Comparative and Robustness Analysis

1) Benchmarking against standard models: To contextualize the performance of the AIRS model, it was benchmarked against three standard machine learning models: a Support Vector Machine (SVM), a Random Forest, and a Neural Network. As shown in Table III, the AIRS model demonstrated highly competitive and, in some metrics, superior performance.

While other models, particularly Random Forest, achieve a slightly higher F1-Score, the AIRS model demonstrates a superior ability to discriminate between classes, as evidenced by the highest ROC AUC score (0.74). This highlights its particular strength for this classification task.

TABLE III. PERFORMANCE COMPARISON OF DIFFERENT MODELS

Model	Accuracy	F1-Score	ROC AUC
AIRS	0.66	0.66	0.74
SVM	0.65	0.68	0.71
Random Forest	0.67	0.70	0.73
Neural Network	0.66	0.69	0.72

TABLE IV. AIRS PERFORMANCE ACROSS DIFFERENT MARKET PERIODS (PART 1)

Period	Accuracy	F1-Score	ROC AUC
Low Volatility	0.70	0.73	0.76
High Volatility	0.65	0.68	0.71

TABLE V. AIRS PERFORMANCE ACROSS DIFFERENT MARKET PERIODS (PART 2)

Period	Accuracy	F1-Score	ROC AUC
Trending Market	0.72	0.75	0.78
Ranging Market	0.64	0.67	0.70

2) Performance across market regimes: The model's robustness was tested across different market conditions. As detailed in Table IV and Table V, the model maintains strong predictive power in trending markets (F1-Score = 0.75) and remains effective in more unpredictable ranging markets (F1-Score = 0.67). Similarly, it performs better in low-volatility periods compared to high-volatility ones, which is an expected and logical outcome.

VI. DISCUSSION

A. Interpretation and Positioning within the Literature

An accuracy of 66.46% on the out-of-sample test set is both statistically and economically significant in a market often approaching a random walk. To properly contextualize this performance, it is essential to compare it to the broader literature, which reveals a clear and crucial dichotomy between statistical accuracy and practical utility. This comparison is detailed in Table VI.

On the one hand, several studies report exceptionally high statistical accuracies. Advanced models like AdaBoost of Bagging (as referenced in Table VI, Paper 1), SVM with a rolling window (Paper 4), and LSTM networks (Paper 5) have shown the ability to achieve accuracies well above 90%. On the surface, these results seem to overshadow the performance reported in this study.

However, critical research provides a more nuanced perspective. Khan et al. (Paper 2) powerfully demonstrate that a model with lower accuracy can be significantly more profitable, establishing that financial backtesting is a more critical standard for validation than raw accuracy. This finding directly challenges the practical relevance of the high-accuracy claims in Papers 1, 4, and 5. Furthermore, the complete failure of certain approaches, like the sentiment-based Stacking Regressor (Paper 3), which performed worse than a naive baseline, highlights the danger of relying on flawed methodologies or incomplete evaluation metrics.

TABLE VI. A COMPARATIVE BENCHMARK OF MACHINE LEARNING APPROACHES FOR STOCK MARKET FORECASTING

Paper	Model	Key Insights	Performance Metrics	Comparison with Literature
1	AdaBoost of Bagging [24]	Hierarchical ensembling (boosting over bagging) improves model robustness and reduces generalization error across multiple metrics.	91.45% (Mean Accuracy) 0.9728 (Mean AUC)	Strong statistical performance, but lacks financial backtesting. The model's practical utility remains unproven, a key limitation highlighted by Paper 2.
2	Random Forest [2]	Critiques the use of accuracy alone. Proves a lower-accuracy model can be more profitable. Establishes financial backtesting as a critical standard for practical validation.	+189.66% Cumulative Return	Sets the benchmark for practical evaluation. This study's financial results provide a critical lens through which the high-accuracy claims of Papers 1, 4, and 5 should be viewed.
3	Stacking Regressor [25]	Demonstrates that sentiment-only regression models are fundamentally flawed, performing worse than a naive baseline, despite having low relative error.	Negative R^2 (e.g., -61.593)	Serves as a critical failure case. The negative R^2 proves the model is unreliable, showing the danger of relying on error metrics like MAPE without a more rigorous metric.
4	Support Vector Machine (SVM) [5]	A rolling window training method dramatically enhances predictive accuracy and stability, but the model's practical profitability remains unproven.	92.48% (Average Accuracy)	High statistical accuracy, but lacks financial validation. As shown in Paper 2, a >90% accuracy does not guarantee profitability.
5	LSTM Network [1]	Validates the high effectiveness of LSTM deep learning models for trend forecasting, but focuses solely on statistical accuracy rather than financial performance.	>93% (for most stocks); up to 97.7%	High accuracy, but methodology is not compared to simpler models. Paper 4 achieved similar scores with a less complex SVM model. Presents a realistic and defensible performance.
6	Artificial Immune Recognition System (AIRS)	Robust predictive capabilities through adaptive learning and evolutionary optimization. Balances accuracy with other critical metrics (Precision, F1, AUC).	68.31% (Validation), 66.46% (Test)	Unlike studies with unproven >90% accuracy, AIRS's performance is comparable to validated, profitable models and avoids the fundamental failures seen in Paper 3.

In this complex landscape, the AIRS model's validated performance of approximately 66.5% is positioned as both realistic and scientifically defensible. It avoids the trap of unproven high-accuracy claims (Papers 1, 4, 5), it is grounded in a robust methodology that avoids the fundamental failures of flawed approaches (Paper 3), and its performance level is situated within a context where practical validation is prioritized over statistical perfection (Paper 2).

B. Practical Implications

The practical implications for traders and financial institutions are significant. The model can be integrated into algorithmic trading systems to generate buy/sell signals. Its ability to provide probabilistic-like insights (via affinity scores) can aid in risk management, and its efficiency, as shown by the cumulative gain curve (Fig. 2), makes it a valuable tool for focusing on high-probability trades.

C. Limitations and Future Directions

Despite its promising performance, this study has several limitations. The model's computational complexity could be a challenge for very high-frequency applications. Its performance is also highly dependent on the quality of the feature engineering and is sensitive to hyperparameter tuning.

Given that this study represents the first successful application of an optimized AIRS model to the foreign exchange market, future work should focus on validating and leveraging its potential in a real-world environment. A crucial next step would be to test its robustness and profitability under simulated or live trading conditions. Such practical validation is essential before considering more complex extensions, such as the integration of alternative data sources or the development of hybrid models that combine AIRS with other machine learning techniques.

VII. Conclusion

This study sought to answer a fundamental question: Can a bio-inspired algorithm like the Artificial Immune Recognition System (AIRS) effectively predict the directional movement of the EUR/USD market? Based on the findings of this research, the answer is a qualified and encouraging "yes".

Following a rigorous process of optimization and validation, the proposed AIRS model achieved an accuracy of 66.5% and a ROC AUC of 0.74 on the out-of-sample test set. While these metrics do not reach the >90% levels sometimes claimed in the literature, they are realistic, statistically significant, and, most importantly, were achieved using a transparent and scientifically defensible methodology. The model proved to be highly competitive against standard benchmarks such as SVM and Random Forest, demonstrating a particular strength in its ability to discriminate between classes.

This work represents an important first step, but it is not without limitations. The model is computationally intensive, and its success relies heavily on the quality of the engineered features. The next logical step in this line of research is to assess the model's profitability in a simulated or live trading environment, as strong statistical performance does not always guarantee positive financial returns.

Nonetheless, the results of this study strongly suggest that Artificial Immune Systems deserve greater attention within the financial forecasting toolbox. They offer a robust and adaptive approach that holds significant potential. It is hoped that this study will encourage further exploration of these powerful algorithms in the field of computational finance.

Acknowledgment

I would like to express my sincere gratitude to Dr. RAOUYANE Brahim, Dr EL MOUMEN Samira, and Dr BELLAFKIH Mostafa for their invaluable assistance and support throughout this work. Their guidance, encouragement, and contributions have played a significant role in the completion of this article.

References

[1] T. Phuoc, P. T. K. Anh, P. H. Tam, and C. V. Nguyen, “Applying machine learning algorithms to predict the stock price trend in the stock market – The case of Vietnam,” Humanit Soc Sci Commun, vol. 11, no. 1, p. 393, Mar. 2024, doi: 10.1057/s41599-024-02807-x.

[2] A. H. Khan et al., “A performance comparison of machine learning models for stock market prediction with novel investment strategy,” PLoS ONE, vol. 18, no. 9, p. e0286362, Sep. 2023, doi: 10.1371/journal.pone.0286362.

[3] T. T. Ngoc, L. V. Dai, and D. T. Phuc, “Grid search of multilayer perceptron based on the walk-forward validation methodology,” IJECE, vol. 11, no. 2, p. 1742, Apr. 2021, doi: 10.11591/ijece.v11i2.pp1742-1751.

[4] A. L. Suárez-Cetrulo, D. Quintana, and A. Cervantes, “Machine Learning for Financial Prediction Under Regime Change Using Technical Analysis: A Systematic Review,” IJIMAI, vol. 9, no. 1, p. 137, 2024, doi: 10.9781/ijimai.2023.06.003.

[5] B. T. Khoa and T. T. Huynh, “Forecasting stock price movement direction by machine learning algorithm,” IJECE, vol. 12, no. 6, p. 6625, Dec. 2022, doi: 10.11591/ijece.v12i6.pp6625-6634.

[6] D. Dasgupta, S. Yu, and F. Nino, “Recent Advances in Artificial Immune Systems: Models and Applications,” Applied Soft Computing, vol. 11, no. 2, pp. 1574–1587, Mar. 2011, doi: 10.1016/j.asoc.2010.08.024.

[7] A. Watkins, J. Timmis, and L. Boggess, “Artificial Immune Recognition System (AIRS): An Immune-Inspired Supervised Learning Algorithm,” Genetic Programming and Evolvable Machines, vol. 5, no. 3, pp. 291–317, Sep. 2004, doi: 10.1023/B:GENP.0000030197.83685.94.

[8] X. Gao, H. Li, H. Zhang, J. Xue, S. Sun, and W. Liu, “Traffic prediction based on spatial-temporal disentangled generative models,” Information Sciences, vol. 680, p. 121142, Oct. 2024, doi: 10.1016/j.ins.2024.121142.

[9] M. Badr and B. Khaled, “O-AIRS: Optimized Artificial Immune Recognition System,” IJCAI, vol. 48, no. 3, Sep. 2024, doi: 10.31449/inf.v48i3.4680.

[10] N. Soltani Halvaiee and M. K. Akbari, “A novel model for credit card fraud detection using Artificial Immune Systems,” Applied Soft Computing, vol. 24, pp. 40–49, Nov. 2014, doi: 10.1016/j.asoc.2014.06.042.

[11] P. Hájek and V. Olej, “Predicting Firms’ Credit Ratings Using Ensembles of Artificial Immune Systems and Machine Learning – An Over-Sampling Approach,” in Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications, vol. 8827, E. Bayro-Corrochano and E. Hancock, Eds., in Lecture Notes in Computer Science, vol. 8827. , Cham: Springer International Publishing, 2014, pp. 29–38. doi: 10.1007/978-3-662-44654-6_3.

[12] K. Polat and S. Güneş, “Medical decision support system based on artificial immune recognition immune system (AIRS), fuzzy weighted pre-processing and feature selection,” Expert Systems with Applications, vol. 33, no. 2, pp. 484–490, Aug. 2007, doi: 10.1016/j.eswa.2006.05.013.

[13] M. El Badaoui, B. Raouyane, S. El Moumen, and M. Bellafkih, "Impact Machine Learning Classification And Technical Indicators, Forecast The Direction Of Bitcoin," in 2023 14th International Conference on Intelligent Systems: Theories and Applications (SITA), Nov. 2023, pp. 1–7. doi: 10.1109/SITA60746.2023.10373706.

[14] V. A. Padilha, V. Magnani, R. C. Gatsios, F. Guasti Lima, and R. Moreira Antonio, "Trend and Momentum Technical Indicators for Investing in Market Indices," EkBis, vol. 8, no. 1, pp. 74–86, Jun. 2024, doi: 10.14421/EkBis.2024.8.1.2163.

[15] A. S. Saud and S. Shakya, "Technical indicator empowered intelligent strategies to predict stock trading signals," Journal of Open Innovation: Technology, Market, and Complexity, vol. 10, no. 4, p. 100398, Dec. 2024, doi: 10.1016/j.joitmc.2024.100398.

[16] M.-Y. Day, Y. Cheng, P. Huang, and Y. Ni, "The profitability of Bollinger Bands trading bitcoin futures," Applied Economics Letters, vol. 30, no. 11, pp. 1437–1443, Jun. 2023, doi: 10.1080/13504851.2022.2060494.

[17] Gollangi, and S. K. Rajaram, "Predictive Analytics for Project Risk Management Using Machine Learning," JDAIP, vol. 12, no. 04, pp. 566–580, 2024, doi: 10.4236/jdaip.2024.124030.

[18] Z. Luo et al., "A Method for Prediction and Analysis of Student Performance That Combines Multi-Dimensional Features of Time and Space," Mathematics, vol. 12, no. 22, p. 3597, Nov. 2024, doi: 10.3390/math12223597.

[19] R. Yacouby and D. Axman, "Probabilistic Extension of Precision, Recall, and F1 Score for More Thorough Evaluation of Classification Models," in Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems, Online: Association for Computational Linguistics, 2020, pp. 79–91. doi: 10.18653/v1/2020.eval4nlp-1.9.

[20] Ž. Đ. Vujovic, "Classification Model Evaluation Metrics," IJACSA, vol. 12, no. 6, 2021, doi: 10.14569/IJACSA.2021.0120670.

[21] D. Chicco and G. Jurman, "The Matthews correlation coefficient (MCC) should replace the ROC AUC as the standard metric for assessing binary classification," BioData Mining, vol. 16, no. 1, Feb. 2023, doi: 10.1186/s13040-023-00322-4.

[22] D. Chicco, M. J. Warrens, and G. Jurman, "The Matthews Correlation Coefficient (MCC) is More Informative Than Cohen's Kappa and Brier Score in Binary Classification Assessment," IEEE Access, vol. 9, pp. 78368–78381, 2021, doi: 10.1109/access.2021.3084050.

[23] Y. Makin and Pavan K Gondhi, "A Quantitative Framework for Portfolio Governance Using Machine Learning Techniques," IJCESEN, vol. 11, no. 3, Jun. 2025, doi: 10.22399/ijcesen.2474.

[24] E. K. Ampomah, Z. Qin, G. Nyame, and F. E. Botchey, "Stock market decision support modeling with tree-based AdaBoost ensemble machine learning models," IJCAI, vol. 44, no. 4, Mar. 2021, doi: 10.31449/inf.v44i4.3159.

[25] S. Elsayed, A. Salah, I. Elhenawy, and M. Abdellah, "Predicting stock prices using ensemble learning techniques," IJECE, vol. 15, no. 2, p. 1783, Apr. 2025, doi: 10.11591/ijece.v15i2.pp1783-1792.

Adaptive Ensemble Models for Robust Intrusion Detection in Cloud Environment on Imbalanced Dataset

Swarnalatha K[1], Nirmalajyothi Narisetty[2]*, Gangadhara Rao Kancherla[3],
Neelima Guntupalli[4], Simhadri Mallikarjuna Rao[5], Archana Kalidindi[6]
Dept. of CSE, Acharya Nagarjuna University, Guntur, 522510, India[1, 3, 4]
Dept. of CSE, Koneru Lakshmaiah Education Foundation, Bowrampet, Hyderabad, Telangana, 500043, India[2]
Dept. of IT, Vasireddy Venkatadri International Technology, Nambur, India[5]
Dept. of CSE-AIML & IoT, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad, India[6]

***Abstract*—The rapid development of Information storage and sharing technologies brings new challenges in protecting against network security attacks. In this study, ensemble learning models are evaluated to enhance the performance of a network intrusion detection system (NIDS) with three phases through machine learning approaches. In the first phase, the unbalanced dataset is processed through four re-sampling techniques, such as SMOTE, RUS, RUS+ROS, and RUS+SMOTE, for balancing treatment. In the second phase, Random Forest feature selection is imposed for these four balanced datasets. Finally, three Ensemble Models named as EM1, EM2 and EM3 are designed using six basic classifiers and thus evaluated. In earlier studies, the first and second phases were evaluated through an SVM binary classifier for four feature subsets. The four feature subsets are obtained through Random Forest feature selection with the four different thresholds of Cumulative Feature Importance Scores (CFIS) (85%, 90%, 95% and 99%). With the observation of the evaluated results, three challenges were identified: i) The highest accuracy obtained through the re-sampling method required maximum computational time. ii) Different thresholds of CFIS exhibit instability in performance metrics as well as computational times, even though the number of features is less. iii) The adopted multi-class SVM classifier's efficiency to detect the attacks within minimum computational time and without compromising accuracy when compared to earlier works is yet to be ascertained. In this study, an attempt has been made to address these challenges with ensemble learning. Three ensemble models are chosen for the evaluation process conducted on the adopted CIIDS -2017 dataset. Finally, the comparative results are presented, and decisive discussions are carried out for implementing the prevention and mitigation algorithms by security professionals.**

Keywords*—Resampling methods; cloud computing; feature selection; ensemble model; intrusion detection system; machine learning

I. INTRODUCTION

The rapid evolution of the cloud computing technology facilitates the infrastructure, network and applications through Internet in the present era. The widespread of cloud computing usage has reduced the management and maintenance costs of IT infrastructure. Cloud resources are managed by different organizations in the network using standards and methods [1]. This has driven its efficient business model and scalability, and at the same time has introduced challenges in network security. Due to its decentralized structure and centralized management, IT infrastructure is vulnerable to contemporary attacks [2].

The biggest obstacle to the success and use of cloud computing by an organization or individual is its security. The most common serious threats in cloud attacks are Distributed Denial of Service (DDoS), which causes excessive volume of traffic from multiple distributed systems, leading to denial of service to users. In DDoS attacks, unauthorized actions aim to temporarily or permanently disrupt the services of network members against counterattacks such as cloud services, thus encouraging the use of cloud computing [3]. The impacts of a successful DDoS attack are varied and can be severely damaging. Organizations may incur significant financial losses from service interruptions, as well as expenses related to incident response and system restoration efforts. Finally, DDoS attacks can cause customers to lose trust in certain products, affecting performance and brand reputation [4] [5].

To maintain the availability and reliability of network services, the research community and industry sectors dedicate significant efforts to developing Intrusion Detection Systems (IDS). Ongoing research by the Cloud Security Alliance (CSA) has identified DDoS attacks as some of the most prevalent threats to cloud security [6]. For example, prominent studies referenced in [7] [8] have concentrated on developing advanced strategies to counter these attacks effectively. Despite the widespread adoption of IDS, DDoS attacks are often avoided in many scenarios. With the growing volume of data constantly transmitted between networks, IDS has proven effective in detecting disruptions within large datasets [9].

Many researchers are working on this to provide a solution in identifying these types of attacks in network intrusion detection based on the available static datasets. Most of these datasets are skewed. Class imbalance in cloud security attacks can influence the performance of intrusion detection systems (IDSs). Since legitimate traffic samples are far beyond malicious, machine learning models are often affected by these groups, resulting in lower detection rates for several types of attacks. This flaw limits the ability of IDSs to detect

rare but important threats, thwarts attack plans, and compromises the security standards of the entire system. Not only in Intrusion detection, other areas like Natural Language Processing [10], [11], Image recognition [12], genetic engineering [13], [10], financial fraud detection [14], web mining to text categorization [15] also have been advocated imbalanced data classification. Therefore, addressing class imbalance is crucial to improve detection accuracy and strengthen defense mechanisms in cloud environments. The quality of the dataset is very significant in the classification method, and certain imbalance classes dominate the dataset.

Ensemble methods proven to be effective in increasing the efficacy of intrusion detection systems (IDS). These methods combine the strength of several machine learning models to achieve improved predictive performance and robustness compared to individual models. Combining various techniques improves the detection by aggregating multiple instances. It also reduces the Variance and biases associated with the limitations of a single integration, especially in high dimensions and skewed data [16] [17].

After going through the above research works, the three-phase ensemble models are designed and evaluated. The coordination methods, such as weighted voting, can mitigate this problem by evaluating the impact of a limited class, thereby reducing the negativity and improving the detection of less resilient types. In addition, to combine re-sampling methods, feature selection with traditional machine learning techniques may be generalize and effective.

The results in [18] demonstrate the performance of re-sampling methods for NIDS. RUS+SMOTE achieves the smallest computational time but delivers the lowest performance. SMOTE achieves the highest accuracy, precision, recall, and F-measure across all CFIS thresholds. In spite of its high accuracy of 99.26% and F-Measure of 91, the (RUS+ROS) is less efficient due to its time complexity of 4808 sec.

To enhance the performance of [18], the current study, which uses ensemble learning, aims to develop a stable and generalized model with low time complexity for real-time imbalanced datasets. The study has been carried out with the following key contributions:

- Study the different re-sampling methods for imbalance treatment to enhance the classification results.
- To develop and evaluate the ensemble classification models from base classifiers for improving the performance of minority class samples.
- To analyze the generalized three-phase ensemble models' behavior in mitigating the class imbalance and improving the AUC.
- Different ensemble models built by the same basic classifier set for comparative analysis.

The rest of the study is organized as follows: Section II provides a summary of the various research efforts in this field. A description of the methodology of the proposed ensemble models is presented in Section III. Section IV explains the analysis of the experimental results. Finally, Section V concludes the current study and discusses future perspectives.

II. Related Work

There have been various studies that have led to the development of automatic intrusion detection in network communication. Various Machine Learning and Deep Learning methods were used to generate the hybrid and ensemble models for dealing with unbalanced data to detect these types of attacks. Some of them are outlined below.

A survey conducted in [19] examined the various strategies for addressing imbalanced data when detecting network intrusions. The common strategies that can be applied to balance the instances and various approaches that can be used to mitigate the challenges of an imbalanced dataset, such as data level methods, algorithm level methods and fusion methods, were explored. The commonly used oversampling techniques in the literature, like SMOTE and ADASYN, were considered. The findings indicate that ADASYN generally outperforms SMOTE in terms of F1-score and recall, while SMOTE may be suitable for maintaining high accuracy.

The authors in [20] aimed to address the challenge of high false negative rates and improve the predictability of minority classes in NIDS for both binary and multiclass classification. There are three stages in the proposed strategy: fine-tuning the training and testing subset distributions, selecting features, and using class weights. The experiments were conducted using the well-established NSL KDD and UNSW-NB15 datasets for binary classification as well as multi-class classification. Multiple models were generated using an effective refinement strategy aimed at reducing False Negative Rates (FNR) and increasing minority class predictability. Based on the results, it was demonstrated that it is possible to achieve a satisfactory trade-off between FNR, accuracy, and minority detection with the proper parameters. While the model's performance on a limited dataset raises concerns related to generalization. This can be addressed in future by exploring the efficacy of the model on large datasets in enhancing the performance on minority attack detection.

Hongpo Zhang et al. proposed a two-stage intrusion detection model for IoT security [21]. Stage 1 employs LightGBM for efficient initial classification, while Stage 2 uses CNN for fine-grained anomaly detection. The model effectively addresses class imbalance and achieves high accuracy, F1-score, and MCC. It outperforms state-of-the-art methods in terms of efficiency and making it a promising solution for large-scale data. Future work will focus on improving feature selection and imbalance handling techniques.

The authors in [22] aimed to address the challenge of high false negative rates and improve the predictability of minority classes in NIDS. In their approach, the training and testing subset distributions are adjusted, features are selected, and class weights are used. Both binary and multi-class classification experiments were conducted with the NSL KDD and UNSW-NB15 datasets. Several models were developed using an effective refinement strategy to reduce False

Negative Rates (FNR) and increase minority class prediction accuracy. It was demonstrated by the results that proper parameters can be utilized to achieve an acceptable trade-off between FNR, accuracy, and minority detection.

The study conducted in [23] investigates the impact of class imbalance countermeasures on model interpretability for both AI users and experts. Despite their effectiveness in improving prediction performance, many countermeasures often compromise interpretability. Interpretability is only preserved by feature selection and cost-sensitive approaches, according to our experiments. Normally, re-sampling and most classification algorithms cannot be used in settings where gaining knowledge and being able to interpret it are essential. Several guidelines are provided for selecting interpretable countermeasures, and we highlight future research opportunities.

The BMCD algorithm has been proposed for large-scale multiclass intrusion detection datasets in [24]. With BMCD, minority class detection is improved by adapting SMOTE to multiclass situations. On a combined dataset from CICIDS2017, BMCD is found to improve intrusion detection performance over existing methods in addressing class imbalance and addressing class imbalance. With BMCD-balanced datasets, random forest classifiers, in particular, perform significantly better.

Pieter Barnard et al. proposed a two-staged pipeline framework in [25] for robust network intrusion detection. The first stage leverages a powerful machine learning model, namely XGBoost. To understand the model's decision-making process, we employ the SHAP framework to generate explanations of its predictions. The second stage utilizes these explanations to train an autoencoder. This autoencoder acts as an anomaly detector, specifically designed to identify unseen attacks not encountered during the initial training. The evaluation is carried out on the NSL-KDD dataset and demonstrates its effectiveness in accurately detecting new attacks. It is also compared to various state-of-the-art intrusion detection methods.

The effect of the class imbalance problem may lead to biased model performance. To address this, the authors Ngan Tran et al. have explored various techniques for handling the class imbalance in NIDS [26]. Among them, the downsampling + upsampling + SMOTE (DUS) was the most effective re-sampling technique based on the experimental results conducted on the NSL-KDD dataset. An Ensemble model combined with DUS outperformed other machine learning classifiers. The impact of the number of classes on model performance is also evaluated, and discovered that more imbalanced classes negatively affect accuracy. Finally, researchers highlighted the importance of addressing class imbalance in NIDS and provided insights into the effective techniques. By understanding the impact of class imbalance and employing appropriate strategies, researchers can improve the performance of NIDS models in real-world scenarios.

In [27], the authors introduced a novel DNN-based approach for detecting DoS/DDoS attacks, which utilizes three imbalanced datasets of NIDS: CICIDS2017, CSE-CICIDS2018, and CICDDoS2019. To mitigate class imbalance, a K-means based technique is employed to generate semi-balanced datasets. Feature selection is performed using Linear Discriminant Analysis, and four metaheuristic algorithms (AIS, FA, IWO, and CS) are integrated with DNN to enhance performance. Experimental results demonstrate the effectiveness of the proposed approach, particularly AIS-DNN, which achieves high accuracy (up to 99.99%) and outperforms existing methods.

According to [28], the quality of the dataset is very important to improve minority class attack detection in NIDS. In this study, a combined oversampling and undersampling technique is used, followed by training with deep learning models. The proposed approach significantly enhances the detection rate of minority classes, with the CNN model achieving 99.8% accuracy in binary classification and the MLP model achieving 99.9% accuracy in multi-class classification. These results offer a promising direction for improving NIDS and detecting minority class attacks.

The authors in [29] demonstrated a novel autoencoder-based anomaly detection method for IoT security. By effectively utilizing packet metadata and training on normal data, the method accurately detects anomalies in encrypted IoT traffic. It outperforms traditional methods, is versatile, and can handle various encryption protocols, making it a robust solution for IoT security. However, the authors have evaluated the performance on only one type of attack.

The Data Generative Model to detect anomalies in an imbalanced dataset was proposed in [30]. Also concentrated on certain types of attacks where those were underrepresented. To address this, the present study proposed a Data Generative Model (DGM) using Conditional Generative Adversarial Networks (CGAN) to increase the number of minority class samples in the CICIDS2017 dataset. The researcher concluded that their experiments demonstrated that the DGM effectively detected new attacks and significantly improved the weighted F1-score (99.12%) compared to existing methods.

Traditional intrusion detection systems often struggle to effectively identify rare but critical attacks due to class imbalance in network traffic [31]. This research introduces a novel framework, S2CGAN-IDS, to address this challenge. The framework categorizes network traffic into three imbalance levels: ample, scarce, and rare. To augment the dataset, it combines the Synthetic Minority Over-sampling Technique (SMOTE) and a novel Synthetically Controlled Generative Adversarial Network (SCGAN). SMOTE oversamples scarce class data, while SCGAN generates synthetic samples for the rare class. The augmented dataset is then fed into a simple Deep Neural Network (DNN) classifier, which effectively distinguishes between normal and anomalous traffic, including rare attacks.

The authors in [32] explained the importance of safeguarding the network to avoid malicious access. Recent advancements in deep learning have significantly improved NIDS performance. However, the inherent class imbalance in network traffic, where normal traffic significantly outweighs attack traffic, remains a significant challenge. To address this issue, this research proposes a hybrid approach that combines oversampling and under sampling techniques. Specifically,

Synthetic Minority Over-sampling Technique (SMOTE) is employed to increase the representation of minority classes, while Tomek Links is used to remove noisy samples. Additionally, two powerful deep learning models, Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN), are utilized to enhance the model's ability to capture intricate patterns in network traffic.

The imbalanced data in NIDS can pose significant challenges, which may lead to biased models that struggle to accurately classify minority classes. Intrusion Detection Systems (IDS) datasets often exhibit this imbalance, with common attacks like DDoS dominating the data. To address this issue, techniques like SMOTE and ADASYN can be used to create synthetic data for minority classes [34]. However, not all features within a dataset are equally important. Feature selection methods like Recursive Feature Elimination (RFE) can help to identify the most relevant features, improving model performance and reducing computational costs. Finally concluded that the Decision Trees consistently outperformed Random Forest and KNN in terms of intrusion detection.

III. Proposed Approach

This section explains the various steps involved in the methodology implementation of the proposed three-phase ensemble model, which is presented in Fig. 1. They are: i) data preparation and imbalance treatment, ii) Random Forest Feature selection and iii) Base classifiers evaluation and evaluating ensemble models. These phases are explained in detail below. The base classifiers are explained briefly in the next subsection.

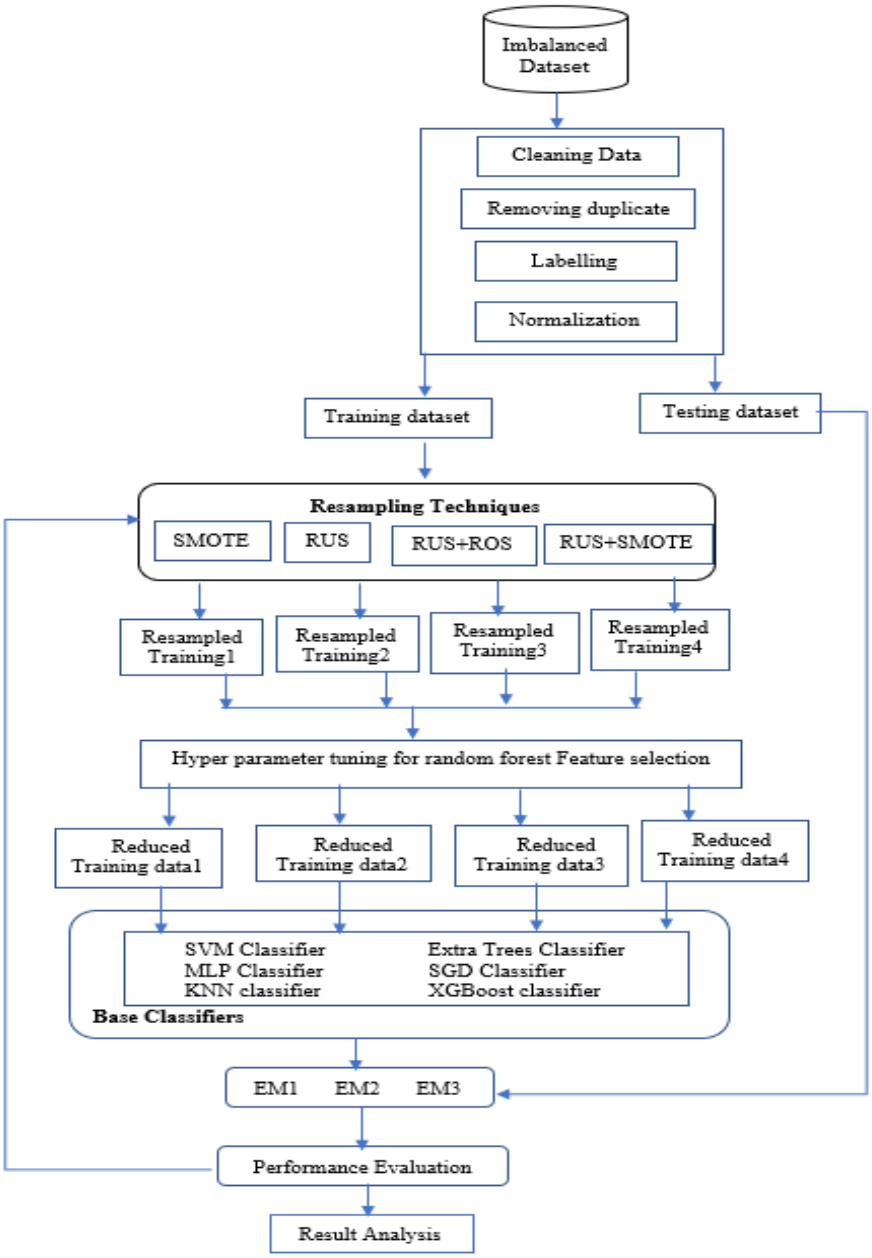


Fig. 1. Methodology of the three-phase ensemble model.

A. Base Classifiers

The construction of three ensemble models, EM1, EM2 and EM3, with the adoption of six base classifiers (BM).

1) Extra Tree classifier: The ET classifier excels in decision making by using the random splits threshold instead of the optimal split. In addition to speeding up the training process, this approach enhances the model's generalization capabilities, reducing overfitting. A key advantage of the Extra Trees Classifier is its ability to produce stable and accurate predictions by averaging outputs from multiple trees. These benefits, combined with its scalability and fast training time, make the Extra Trees Classifier a powerful and reliable tool in intrusion detection [35], and the graphical representation is shown in Fig. 2. It also still has the limitation that it overfits if the dataset contains too much noise.

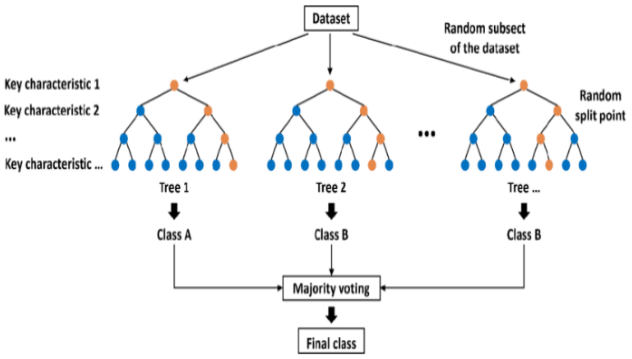


Fig. 2. Extra Tree Classifier.

By optimizing a loss function using gradient descent, it is known for its scalability and effectiveness [36].

2) Support Vector Machine (SVM): It is one of the popular classifiers in machine learning and is well known for its effectiveness in high-dimensional spaces as well as with non-linear decision boundaries. It excels in identifying the patterns among the data samples using various non-linear kernel functions by preventing the overfitting problem. Based on the assumption that all the above advantages will help to improve the performance of intrusion detection systems, the SVM has also been included in the investigation [37]. The following Fig. 3 shows the pictorial representation of the SVM classifier.

3) MLP classifier: It is a type of neural network and is capable of capturing complex, non-linear relationships in data. This makes it particularly valuable for detecting intricate patterns in the data, which are common in intrusion detection scenarios. The foundation of a neural network is a perceptron, which contains three layers of neurons, namely, the input layer, hidden layer and output layer. It uses backpropagation for improving the learning rate and decreasing the error rate [38]. The MLPs are flexible and can model intricate decision boundaries, providing additional power to the ensemble model.

4) KNN classifier: Because of its many benefits, the KNN classifier is often chosen for machine learning jobs. Its non-parametric nature enables it to handle datasets with irregular patterns efficiently, and its simplicity and ease of implementation make it perfect for learners. KNN is adaptable, works well for both regression and classification applications, and naturally manages multi-class issues. It doesn't need a training phase because it is a lazy learner, which makes it computationally efficient for creating models. It works well with tiny datasets and, by taking into account the local structure of the data, can manage non-linear decision limits. KNN also doesn't depend on presumptions about feature interactions and is resilient to outliers (if k is chosen correctly).

5) XG boost: With so many benefits, the XGBoost classifier is a popular and powerful machine learning technique. With features like tree pruning and parallel processing that condense computation times and improve scalability, it is incredibly effective and performance-optimized. XGBoost produces strong and trustworthy predictions even when dealing with big datasets and missing values. It is perfect for complex models since regularization techniques like L1 and L2 can reduce overfitting. It is appropriate for a variety of applications, such as recommendation systems, healthcare, and finance, due to its high performance, speed, and adaptability.

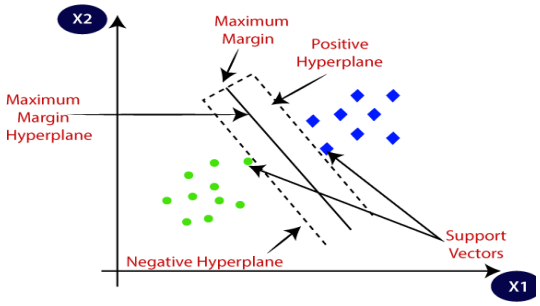


Fig. 3. Pictorial representation of an SVM classifier.

The main objective of selecting the above classifiers is to benefit from the diverse learning philosophy in building a robust generalized model towards improving the intrusion detection rate [33].

B. Phase 1

This subsection consists of two stages: Data preparation and Data imbalance treatment.

1) Data preparation: The preliminary stage of developing an Intrusion Detection System (IDS) includes gathering comprehensive data on network traffic that covers both benign and contemporary attacks. These types of attacks are resembled in the datasets CICIDS-2017. The complete data preparation steps, like data cleaning, eliminating duplicate rows and Label encoding, are carried out in line with [19]. A normalization step is performed to standardize feature magnitudes, enhancing the algorithm's robustness and ensuring no single feature dominates during learning.

2) Data imbalance treatment: Data imbalance in intrusion detection systems (IDS) is a critical issue that affects the system's ability to detect malicious activity effectively. Intrusion detection datasets often exhibit a significant imbalance, where normal (benign) activities vastly outnumber malicious (attack) events. Treating this imbalance is essential to improve the system's performance and reliability. The main advantages of balancing the dataset are enhancing model generalization, improving the detection rate and minority sample detection. To balance the class labels, the study utilizes four resampling methods, SMOTE, RUS, RUS+ROS and RUS+SMOTE, outlined by [34] and [18]. All four methods were applied to training data in parallel to generate respective balanced datasets.

C. Phase 2

In this phase, feature selection is carried out with the adoption of the Random Forest feature selection method as per the feature extraction process and hyperparameter tuning given in [18]. The purpose behind selecting different thresholds in the progressive approach is to conduct a systematic study of the balance between complexity, robustness and accuracy. The resulting datasets are fed as input to the feature selection phase to mitigate the overfitting problem and carried out based on CFIS score thresholds (85%, 90%, 95% and 99%) for extracting four feature subsets.

D. Phase 3

This classification phase consists of six base classifiers' evaluation, along with three ensemble models' evaluation.

1) Base classifiers evaluation: The results of the study [18] in terms of intrusion detection were promising, but there is still scope for optimizing the computational complexity as well as the performance to generate a stable model. The current study is conducted with the objective of addressing these aspects by building a generalized ensemble model. The four different datasets generated as output of feature selection phase are given as input to four base classifiers to generate each different ensemble model based on majority voting. The novelty of the present study generates an ensemble model with different datasets contrasting to the existing ensemble models, where all the base classifiers will be training with same dataset. Each time four classifiers are selected from the following base classifiers Extra Trees (ET), Stochastic Gradient Descent (SGD), Support Vector Machine (SVM), KNN, XGBoost and Multi-Layer Perceptron (MLP) to build the ensemble models.

2) Evaluation of EM1, EM2 and EM3: The EM1 model is constructed with Extra Tree, SGD, SVM, and XGBoost Classifiers. EM2 model created with KNN, SGD, SVM, and MLP Classifiers. The EM3 is built with Extra Tree, SGD, SVM, and MLP Classifiers. The strategic decision to select these six diverse base classifiers aimed at leveraging their complementary strength to generate more robust and accurate classifiers in classifying the network samples. The strengths or the advantages of these algorithms are discussed below. Voting is an ensemble learning technique that works better when the predictions are combined for a classification case. Much better performance is typically obtained by voting rather than using a single classifier [39].

Two kinds of voting are used in classification problems — hard voting [40] and soft voting [41]. Hard voting is to take the one with the most votes, while soft voting means averaging probabilities. The current study leveraged soft voting; its importance stems from its ability to improve prediction accuracy and robustness by leveraging the strengths of multiple models. The final prediction is based on the weighted average of these probabilities, leading to a more nuanced decision-making process. The output of the snippet for ensemble mode implementation is shown in Fig. 4. Finally, the pseudocode of the three-phase ensemble model is presented in Fig. 5.

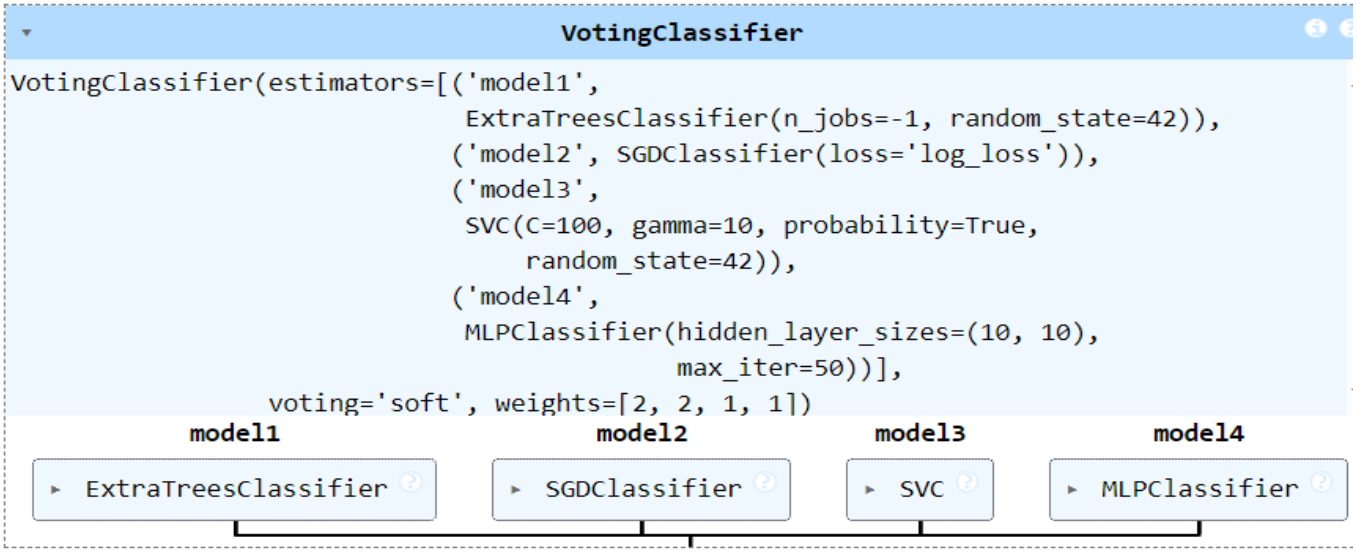


Fig. 4. Snippet of the ensemble classifier from python implementation.

```
Input: unbalanced dataset D={(x1,y1),(x2,y2),……..,(xn,yn)} when yi=[0,1,….k] and k= number of classes
Output: Ensemble Classifier
Cumulative Feature Importance Score: CFISthreshold=[85%,90% 95%,99%]

        Step 1:Preprocessing
                Step 1a: for each categorical Features Fi in D do
Fi<-- Labelling
end for
        Step 1b: Remove the duplicate instances
        Step 1c: Eliminate Feature Fi with statistical measure 0
        Step 1d: Divide dataset into train(Dtrain) and test(Dtest) set

Step 2: BM<- {SMOTE, RUS, RUS+ROS, RUS+SMOTE}
                Define the Sampling strategy
                Select Y min<- minority class
for each BMi generate synthetic instances of Y min as balance dataset DBtrain,DBtest

        Step 3:
Step3a: impscore<-- calculateScore(feature_list,RF, DBtrain, DBtest)
                Step 3b: imp_feature_list<-- selectFeature(feature_list, impscore,CFISthreshold)
                Step 3c: get reduced features DBRtrain, DBRtest

        Step 4: voting= ”soft”
Step 4a: selected Fi in DB do
M1=ET(DBRtrain, DBRtestTrain_label)
M2=SGD(DBRtrain, DBRtest ,Train_label)
M3=SVM (DBRtrain, DBRtest ,Train_label)
M4=MLP(DBRtrain, DBRtest ,Train_label)
M5=KNN(DBRtrain, DBRtest ,Train_label)
M6=XGBoost(DBRtrain, DBRtest ,Train_label)

Step 4b: Ensemble_Modeli(DBRtrain, DBRtest,Train_label)
soft_voting_classifier=Concatenate(M1,M2,M3,M4)
        soft_voting_classifier.fit(DBRtrain,Train_label)
predictions=soft_voting_classifier.predict(DBRtest)

Step 5: Result
```

Fig. 5. Pseudocode of the proposed methodology.

IV. Results and Discussion

In this section, the experimental results are analyzed, and the pros and cons of the proposed ensemble models based on the widely used classification metrics are given. The metrics are Accuracy, Precision, Recall, F-Measure and Time complexity. Besides that, another evaluation metric, AUC-ROC, is also chosen for measuring ensemble model classification performance. The EM1, EM2 and EM3 models' evaluation results are presented in Table I, Table II and Table III, respectively. They are obtained from experiments conducted on the CICIDS-2017 benchmark dataset after applying four types of imbalance treatments with random forest feature selection.

Based on these results, the ensemble model consistently achieves 99.88% accuracy across all thresholds (85%, 90%, 95%, 99%). A strong balance between robustness and quality is evident in the precision of 99.67%, recall of 99.58%, and F-Measure of 99.62 %. In terms of time complexity, this model has a low range of 299.572 to 362.86, making it very efficient. By combining these two classifiers, patterns in the data are captured effectively while remaining computationally efficient. As it exhibits high accuracy and reliability at all thresholds, it is an ideal choice for tasks that require precision and speed at the same time. The consistent performance across different thresholds suggests that the model is stable and does not overfit. In spite of its low time complexity, it is scalable for large datasets due to its low time complexity. Overall, this ensemble is a top-performing model and well-suited for real-time applications.

In comparison to the above scenario, this ensemble model delivers slightly lower accuracy, ranging from 99.4% to 99.51%. In this case, precision remained around 99.18%, but recall dropped to 93.52, resulting in a lower F-Measure of approximately 95.74 to 95.79%. The primary downside of this model is its high time complexity at the 90% threshold, which spikes to 2128.69, making it impractical for real-time applications. In contrast, as the threshold increases, the time complexity drops significantly, going from 2128.692 to 555.78 for higher thresholds. This model is uneconomical in comparison with the first scenario because it has a high computational cost for the lower thresholds, despite its reasonable accuracy and F-measure. Small datasets or non-real-time tasks are suitable for this ensemble when time complexity is less of an issue. Overall, the program demonstrates good prediction capabilities, but it doesn't perform well in terms of efficiency.

TABLE I. Effect of Various Performance Metrics for EM1 Model

CFIS	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)	Time complexity(sec.)
85%	99.88	99.67	99.58	99.62	299.572
90%	99.88	99.67	99.58	99.62	313.63
95%	99.88	99.67	99.58	99.62	362.86
99%	99.88	9967	9958	99.62	321.58

TABLE II. EFFECT OF VARIOUS PERFORMANCE METRICS FOR EM2 MODEL

CFIS	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)	Time complexity(sec.)
85%	99.51	99.18	93.51	95.79	571.62
90%	99.4	99.09	93.51	95.74	2128.692
95%	99.51	99.04	93.51	95.77	499.52
99%	99.43	99.14	93.51	95.77	555.78

TABLE III. EFFECT OF VARIOUS PERFORMANCE METRICS FOR EM3 MODEL

CFIS	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)	Time complexity(sec.)
85%	99.6	99.6	99.6	99.6	363.77
90%	99.51	99.51	99.51	99.51	675.66
95%	99.6	99.6	99.6	99.6	388.81
99%	99.6	99.6	99.6	99.6	380.62

All the parameters of this ensemble remain at 99.6% accuracy, precision, recall, and F-Measure. Compared to the first ensemble, this ensemble has a slightly higher level of time complexity between 363.77 and 675.66, which is still manageable. Stability of the model is evident from its identical metrics across thresholds, which indicates strong generalization and robustness. The time complexity of its ensemble is higher than that of the Extra Tree ensemble, but it remains far more efficient than the KNN ensemble in terms of computational efficiency. The model is a great option for applications that require consistency and balance in performance. It is well-suited for tasks involving high-dimensional data or requiring consistent outcomes. Overall, the accuracy and computing efficiency of this ensemble are well balanced.

It can be observed that the three proposed ensemble models yield the same level of performance without any significant differences. This is true for all CFIS scores. For this reason, an 85% CFIS threshold is chosen to evaluate the AUC-ROC metric and then compared with the same CFIS threshold of results in [18]. The relevant confusion matrices are given below.

There are some noticeable improvements in terms of the classification of class 0 and class 2 instances in the second matrix compared to the first matrix, which is derived from the methodology of [18], where they were misclassified. This may result in a potential model with better separation of the model.

The following analysis is grounded on Fig. 6 and Fig. 7. The performance of the EM1 model on class-0 is exceptionally good, and the rate of misclassification is negligible with respect to the size of the actual class 0 compared to the model [18] performance, where there is a high misclassification. The ensemble model reduces the misclassification samples to 50% in the ensemble model for class-1. Accordingly, class-2, class-3 and class-4 performance is very effective with only 37,11 and 14 number of samples being misclassified, which is a negligible percentage compared to the actual number of samples 69037,1650 and 1739, respectively. Coming to class-5, which is the minor class among all, has classified 100% correctly, whereas it is not in line with the first matrix. Fig. 6 shows that there are significant misclassifications in classes 0 and 2.

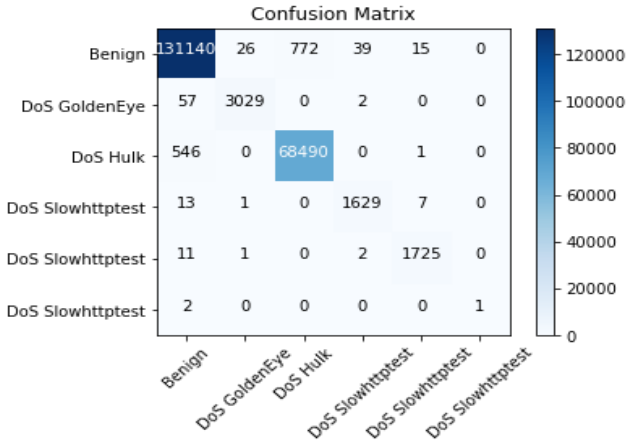


Fig. 6. Confusion Matrix of [18] with CFIS threshold of 85%,

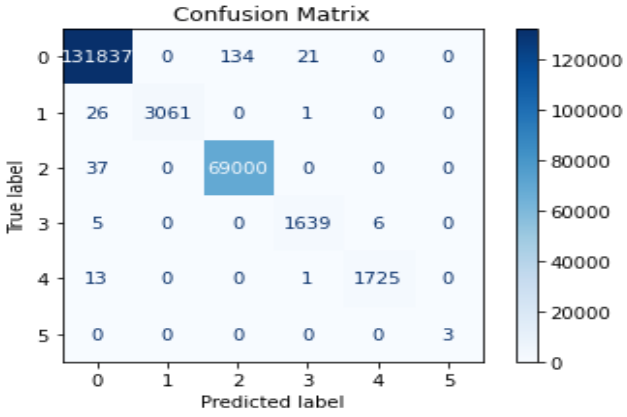


Fig. 7. Confusion matrix of EM1 with CFIS threshold of 85%.

Fig. 8 demonstrates a classifier's performance among six classes in a multi-class Receiver Operating Characteristic (ROC) curve plot. The True Positive Rate (TPR) is represented by the y-axis, and the False Positive Rate (FPR) by the x-axis. Every colored curve represents a distinct class; when all curves reach the upper-left corner, perfect categorization is indicated. Random prediction is represented

by the diagonal dashed line, but the model performs far better. Each class's Area Under the Curve (AUC) is 1.00, indicating perfect performance devoid of misclassifications. These findings point to a remarkable model, but they may also point to possible overfitting, especially if the dataset is tiny or not typical of real-world situations.

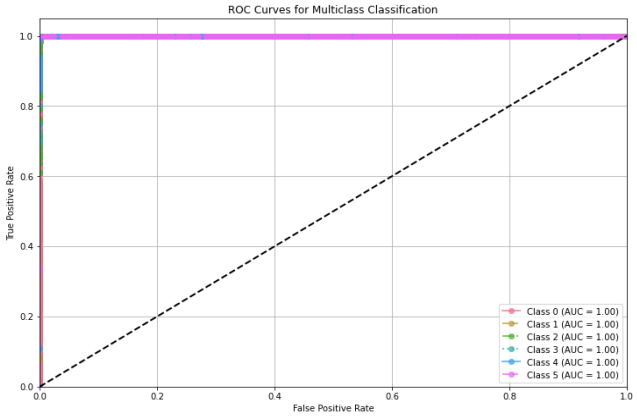


Fig. 8. ROC-AUC curve ensemble model in CICID-2017.

V. Conclusion and Future Scope

In this study, to improve and enhance the performance of NIDS, Ensemble learning is adopted. This technique is used to develop three Ensemble models that are designed in the proposed work. All these models have boosted the performance of the existing method. Each of the proposed Ensemble models has used four classifiers out of six base classifiers, whereas [18] has adopted an SVM binary classifier. It has shown better performance with respect to all the metrics for all chosen re-sampling methods with different CFIS scores. From the analysis of the results of the proposed work, one can observe that an 85% threshold feature subset with a minimum number of features is sufficient for detecting attack classification with minimum computational complexity and without any loss in accuracy. From this process of evaluation, the EM1 model was evaluated with a minimum computational time and gives a better trade-off between computational time and classification metrics. It has been proven that the four re-sampling methods with the EM1 model, along with Random Forest feature selection, outperform. Thus, it is a suggested model for attack detection in real-time, imbalanced traffic of NIDS. The limitation of this study is that the experiments were conducted on a training and testing split only, not executed using cross-validation.

To make the results much clearer, explainable models like SHAPE, LIME can be utilized. As a future scope of this study, statistically based feature selection methods may be chosen for the evolutionary process with different combinations of re-sampling methods or base classifiers. Further, this method can be applied to contemporary datasets for new type of attacks.

References

[1] Y. Sanjalawe and T. Althobaiti, DDoS attack detection in cloud computing based on ensemble feature selection and deep learning, Comput. Mater. Contin., vol. 75, no. 2, pp. 3571–3588, 2023.

[2] M. Ouhssini, K. Afdel, M. Akouhar, E. Agherrabi, and A. Abarda, "Advancements in detecting, preventing, and mitigating DDoS attacks in cloud environments: A comprehensive systematic review of state-of-the-art approaches," Egypt. Inform. J., vol. 27, no. 100517, p. 100517, 2024.

[3] G. S. Kushwah and V. Ranga, "Voting extreme learning machine based distributed denial of service attack detection in cloud computing," J. Inf. Secur. Appl., vol. 53, no. 102532, p. 102532, 2020.

[4] S. Pahal and A. Saroha, "Distributed Denial of Services attacks on cloud servers: Detection, Analysis, and Mitigation," Mapana Journal of Sciences, vol. 22, no. 1, pp. 121–145, 2023.

[5] G. S. Kushwah and V. Ranga, "Optimized extreme learning machine for detecting DDoS attacks in cloud computing," Comput. Secur., vol. 105, no. 102260, p. 102260, 2021.

[6] D. Zeng, J. Zhang, L. Gu, S. Guo, and J. Luo, "Energy-efficient coordinated multipoint scheduling in green cloud radio access network," IEEE Trans. Veh. Technol., vol. 67, no. 10, pp. 9922–9930, 2018.

[7] Y. Mehmood, M. A. Shibli, U. Habiba, and R. Masood, "Intrusion Detection System in Cloud Computing: Challenges and opportunities," in 2013 2nd National Conference on Information Assurance (NCIA), 2013.

[8] S. R. K. Tummalapalli and A. S. N. Chakravarthy, "Intrusion detection system for cloud forensics using bayesian fuzzy clustering and optimization based SVNN," Evol. Intell., vol. 14, no. 2, pp. 699–709, 2021.

[9] F. Palmieri, S. Ricciardi, U. Fiore, M. Ficco, and A. Castiglione, "Energy-oriented denial of service attacks: an emerging menace for large cloud infrastructures," J. Supercomput., vol. 71, no. 5, pp. 1620–1641, 2015.

[10] Y. Li, H. Guo, Q. Zhang, M. Gu, and J. Yang, "Imbalanced text sentiment classification using universal and domain-specific knowledge," Knowl. Based Syst., vol. 160, pp. 1–15, 2018.

[11] R. Panigrahi and S. Borah, "Dual-stage intrusion detection for class imbalance scenarios," Comput. Fraud Secur., vol. 2019, no. 12, pp. 12–19, 2019.

[12] L. Wang and C. Wu, "Dynamic imbalanced business credit evaluation based on Learn++ with sliding time window and weight sampling and FCM with multiple kernels," Inf. Sci. (Ny), vol. 520, pp. 305–323, 2020.

[13] Y. Liu, Z. Yu, C. Chen, Y. Han, and B. Yu, "Prediction of protein crotonylation sites through LightGBM classifier based on SMOTE and elastic net," Anal. Biochem., vol. 609, no. 113903, p. 113903, 2020.

[14] M. E. El-Telbany, "Prediction of the electrical load for Egyptian energy management systems: Deep learning approach," in Advances in Intelligent Systems and Computing, Cham: Springer International Publishing, 2020, pp. 237–246.

[15] Privacy, and Anonymity in Computation, Communication, and Storage: 9th International Conference. SpaCCS; Zhangjiajie, China, 2016.

[16] S. Oyucu, O. Polat, M. Türkoğlu, H. Polat, A. Aksöz, and M. T. Ağdaş, "Ensemble Learning framework for DDoS detection in SDN-based SCADA systems," Sensors (Basel), vol. 24, no. 1, 2023.

[17] K. Alluraiah and M. S. R. Chetty, "Ensemble learning method for DDOS attack mitigation in web based networks," Int. J. Intell. Syst. Appl. Eng., vol. 12, no. 3, pp. 743–753, 2024.

[18] S. Kudithipudi, N. Narisetty, G. R. Kancherla, and B. Bobba, "Evaluating the efficacy of resampling techniques in addressing class imbalance for network intrusion detection systems using support vector machines," Ing. Syst. D Inf., vol. 28, no. 5, pp. 1229–1236, 2023.

[19] K. Swarnalatha, N. Narisetty, G. Rao Kancherla, and B. Bobba, "Analyzing resampling techniques for addressing the class imbalance in NIDS using SVM with Random Forest feature selection," International Journal of Experimental Research and Review, vol. 43, pp. 42–55, 2024.

[20] E. A. Al-Qarni and G. A. Al-Asmari, "Addressing imbalanced data in network intrusion detection: A review and survey," Int. J. Adv.

Comput. Sci. Appl., vol. 15, no. 2, 2024.

[21] J. Mijalkovic and A. Spognardi, "Reducing the false negative rate in deep learning based network Intrusion Detection Systems," Algorithms, vol. 15, no. 8, p. 258, 2022.

[22] H. Zhang, B. Zhang, L. Huang, Z. Zhang, and H. Huang, "An efficient two-stage network intrusion detection system in the internet of Things," Information (Basel), vol. 14, no. 2, p. 77, 2023.

[23] N. D. Patel, B. M. Mehtre, and R. Wankar, "A computationally efficient dimensionality reduction and attack classification approach for network intrusion detection," Int. J. Inf. Secur., vol. 23, no. 3, pp. 2457–2487, 2024.

[24] D. Cemernek, S. Siddiqi, and R. Kern, Effects of Class Imbalance Countermeasures on Interpretability. IEEE Access, 2024.

[25] A. A. Abdulrahman and M. K. Ibrahem, "Toward constructing a balanced intrusion detection dataset based on CICIDS2017," Samarra Journal of Pure and Applied Science, vol. 2, no. 3, pp. 132–142, 2020.

[26] P. Barnard, N. Marchetti, and L. A. DaSilva, "Robust network intrusion detection through explainable artificial intelligence (XAI)," IEEE Netw. Lett., vol. 4, no. 3, pp. 167–171, 2022.

[27] N. Tran, H. Chen, J. Jiang, J. Bhuyan, and J. Ding, "Effect of class imbalance on the performance of machine learning-based network intrusion detection," International Journal of Performability Engineering, vol. 17, no. 9, 2021.

[28] O. Mjahed, S. El Hadaj, E. Mahdi El Guarmah, and S. Mjahed, "New denial of service attacks detection approach using hybridized deep neural networks and balanced datasets," Comput. Syst. Sci. Eng., vol. 47, no. 1, pp. 757–775, 2023.

[29] A. Abdelkhalek and M. Mashaly, "Addressing the class imbalance problem in network intrusion detection systems using data resampling and deep learning," J. Supercomput., vol. 79, no. 10, pp. 10611–10644, 2023.

[30] M.-G. Kim and H. Kim, "Anomaly detection in imbalanced encrypted traffic with few packet metadata-based feature extraction," Comput. Model. Eng. Sci., vol. 141, no. 1, pp. 585–607, 2024.

[31] A. S. Barkah, S. R. Selamat, Z. Z. Abidin, and R. Wahyudi, "Data Generative Model to Detect the Anomalies for IDS Imbalance CICIDS2017 Dataset," Dataset. TEM Journal, no. 1, 2017.

[32] C. Wang, D. Xu, Z. Li, and D. Niyato, "Effective intrusion detection in highly imbalanced IoT networks with lightweight S2CGAN-IDS," arXiv [cs.CR], 2023.

[33] M. Mbow, H. Koide, and K. Sakurai, "Handling class imbalance problem in intrusion detection system based on deep learning," International Journal of Networking and Computing, vol. 12, no. 2, pp. 467–492, 2022.

[34] A. S. Barkah, S. R. Selamat, Z. Z. Abidin, and R. Wahyudi, "Impact of data balancing and feature selection on machine learning-based network intrusion detection," JOIV Int. J. Inform. Vis., vol. 7, no. 1, p. 241, 2023.

[35] P. Goyal, R. Rani, and K. Singh, "Comparative analysis of machine learning and ensemble learning classifiers for Alzheimer's disease detection," in 2022 IEEE International Conference on Current Development in Engineering and Technology (CCET), 2022.

[36] H. M. Saleh and A. Hend Marouane, "Stochastic gradient descent intrusions detection for wireless sensor network attack detection system using machine learning," IEEE Access, vol. 12, pp. 3825–3836, 2024.

[37] N. Nirmalajyothi, K. Rao, B. B. Rao, and K. Swathi, "Performance of Various SVM Kernels for Intrusion Detection of Cloud Environment," International Journal of Emerging Trends in Engineering Research, vol. 8, no. 10, 2020.

[38] A. A. Hagar and B. W. Gawali, "Implementation of machine and deep learning algorithms for intrusion detection system," in Intelligent Communication Technologies and Virtual Mobile Networks, Singapore: Springer Nature Singapore, 2023, pp. 1–20.

[39] S. Džeroski and B. Ženko, "Stacking with multi-response model trees," in Lecture Notes in Computer Science, Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 201–211.

[40] I. Gandhi and M. Pandey, "Hybrid Ensemble of classifiers using voting," in 2015 International Conference on Green Computing and Internet of Things (ICGCIoT), 2015.

[41] S. W. A. Sherazi, J.-W. Bae, and J. Y. Lee, "A soft voting ensemble classifier for early prediction and diagnosis of occurrences of major adverse cardiovascular events for STEMI and NSTEMI during 2-year follow-up in patients with acute coronary syndrome," PLoS One, vol. 16, no. 6, p. e0249338, 2021.

Detection of Leaf Fall Disease in Sembawa Rubber Plantation Through Feature Extraction Model and Clustering Methods

Alhadi Bustamam[1], Devvi Sarwinda[2], Retno Lestari[3], Ahmad Ihsan Farhani[4], Harum Ananda Setyawan[5], Masita Dwi Mandini Manessa[6], Tri Rappani Febbiyanti[7], Minami Matsui[8]

Department of Mathematics-Faculty of Mathematics and Natural Science, Universitas Indonesia, Depok, Indonesia[1, 2, 4, 5]
Data Science Center UI-Faculty of Mathematics and Natural Science, Universitas Indonesia, Depok, Indonesia[1, 2, 4, 5]
Department of Biology-Faculty of Mathematics and Natural Science, Universitas Indonesia, Depok, Indonesia[3]
Department of Geography-Faculty of Mathematics and Natural Science, Universitas Indonesia, Depok, Indonesia[6]
Indonesian Rubber Research Institute, South Sumatra, Indonesia[7]
RIKEN Tsurumi-ku, Yokohama, Kanagawa, 230-0045, Japan[8]

***Abstract*—Natural rubber is one of Indonesia's most important export commodities, making the country the second-largest exporter globally with a 28.65% share of the world market. However, recent production has declined, partly due to leaf fall disease caused by the *Pestalotiopsis sp.* fungus. This disease leads to premature leaf drop, which forces rubber trees to redirect energy from latex production to leaf regeneration, potentially reducing yields by up to 30%. Traditional detection methods that rely on manual visual inspection of leaf morphology are impractical over large plantation areas. To address this, the present study proposes a remote sensing-based detection approach using aerial drone imagery and unsupervised machine learning. Two feature extraction methods: Convolutional Autoencoder (CAE) and Gray Level Co-occurrence Matrix (GLCM) were used prior to clustering with k-means. Despite a small dataset, the GLCM-based approach significantly outperforms the CAE-based method. These results demonstrate that GLCM combined with clustering can reliably distinguish between healthy and diseased plantation areas. The proposed method offers a cost-effective, scalable, and non-invasive alternative to ground surveys, and has strong potential for real-world deployment in disease monitoring and early warning systems across large agricultural regions.**

***Keywords*—Convolutional autoencoder; gray level co-occurrence matrix; k-means clustering; rubber plant plantation; Pestalotiopsis sp**

I. Introduction

The rubber plant (*Hevea brasiliensis*) is a plant in the form of a tall tree and possesses a substantial stem that generates natural rubber. Natural rubber is one among the essential goods for Indonesia to export. From 2014 to 2019, the average export of natural rubber in Indonesia made up to 2.691.120 tons, and Indonesia dominates 28.65% world market share. This makes Indonesia the second biggest natural rubber exporter in the world after Thailand [1].

Recently, natural rubber production in Indonesia has been decreasing. One of the factors causing the decline is the emergence of new leaf fall disease caused by the fungus *Pestalotiopsis sp*. The leaf fall disease causes the rubber tree to drop its leaves prematurely, resulting food reserve of the rubber tree allocated into regrow fallen leaves [2]. It is estimated that the area of rubber plantations in Indonesia was around 3.692.352 hectares in 2021 [3]. The area of rubber plantation that is infected with *Pestalotiopsis sp.* is around 30.328,84 hectares, resulting in a decline of natural rubber production by up to 30% [4].

The disease can be detected through the morphology of the leaf. The visible symptoms on the leaves manifest as brown spots, which subsequently progress into dark brown areas, with a distinct demarcation separating these affected regions from the healthy leaf sections. The size of these spots gradually increases to reach dimensions of 1 to 2 centimeters, eventually leading to necrosis in the surrounding tissue [4].

The ratio of a plant's leaf area to its land surface area is called the leaf area index (LAI) [5]. Research by [6] claims that there is a vigorous negative connection amongst severeness of disease resulting in leaves falling and the leaf zone index. This implies that if the disease is more severe, the leaf area index will also diminish. As the leaf area index decreases, the leaf area is also decreasing, implying that the number of leaves in the tree is also decreasing. When the leaf loss reaches a certain point, the rubber plant will find it difficult to photosynthesis, hence the rubber tree will allocate its food reserves to grow new leaves, rather than using its food reserve to produce latex [6].

When photographed with an aerial drone, the difference between a healthy rubber plantation and an infected rubber plantation is visible. Healthy rubber plantation has dense and tightly closed canopy while infected rubber plantation has sparse and not tightly closed canopy [6]. This gives authors an idea to create a disease detection model using machine learning as an early detection to detect leaf fall disease in rubber plantations particularly caused by the fungus *Pestalotiopsis sp.*, using aerial photos of rubber plantations as the dataset.

Early detection of leaf fall disease offers substantial benefits for rubber farmers and plantation managers.

*Corresponding Author

Identifying affected areas at an early stage enables targeted intervention, such as localized fungicide application or canopy management, which can prevent the disease from spreading further. This minimizes crop loss, preserves yield potential, and reduces unnecessary operational costs associated with broad area treatments. Furthermore, by using aerial imagery and automated analysis, farmers can monitor large plantation areas more efficiently and frequently, improving response times and reducing dependence on labor-intensive ground inspections. Such scalable detection systems can ultimately support more resilient and sustainable rubber production.

Most previous studies on plant disease detection rely on labeled datasets and employ supervised classification methods to identify disease types, typically using close-up images of individual leaves. However, to the best of our knowledge, there is no prior research that utilizes aerial drone imagery to either classify or cluster plantations affected by leaf fall disease. Given that the dataset in this study is unlabeled and collected at the plantation scale using an aerial drone, an unsupervised clustering approach is adopted to detect and differentiate between healthy and infected rubber plantation areas.

Before clustering the infected plantations, the images will be inputted through the feature extraction method. The feature extraction method is used to obtain as much information as possible from the images and to diminish the image's size [7]. The information extracted by feature extraction will then be used as an input for the clustering method to cluster.

The feature extraction used in this research is a convolutional autoencoder (CAE) and gray level co-occurrence matrix (GLCM). Convolutional autoencoder uses a convolution layer and a pooling layer to extract temporal and spatial features from the image [8]. Meanwhile, GLCM uses a matrix, where all elements from the matrix are pairs of pixels that have a brightness level, distance d, and inclination angle θ [9]. These two approaches are applied to cut down the dimension of the picture and extract prominent characteristics from the images.

In this research, the authors use CAE and GLCM as feature extraction methods to cluster healthy rubber plantations and infected rubber plantations, particularly caused by the fungus *Pestalotiopsis sp.* K-means clustering will then be used as a clustering algorithm. Convolutional autoencoder and GLCM are used to reduce the dimension of images and extract prominent features. Then the extracted features will be clustered using k-means clustering, and finally, the result will be evaluated using the silhouette score and davies-bouldin index.

The major contributions of this study are as follows:

- A novel application of unsupervised clustering (k-means) to distinguish between healthy and diseased rubber plantations using aerial drone imagery, without requiring labeled data.
- Implementation and comparison of two feature extraction techniques: CAE and GLCM to identify prominent image characteristics for clustering.
- Empirical validation showing that the GLCM-based approach outperforms CAE in silhouette score and davies-bouldin index, highlighting its suitability for small-sample scenarios.
- A cost-effective, scalable detection framework for early disease identification that supports better decision-making and operational efficiency in large-scale rubber plantations.

This research is organized into five sections: Section I provides the background and motivation for the study. Section II reviews related works and previous research relevant to this topic. Section III describes the dataset, methodology, and models employed in the study. Section IV presents and discusses the results. Finally, Section V concludes the research and outlines potential directions for future work.

II. Related Work

This section outlines previous research related to the current study. A number of past investigations have focused on detecting plant disease using machine learning, and a summary of these studies is presented in TABLE I

TABLE I State-of-the-Art

Research	Method	Dataset	Evaluation
[10]	MobileNetV3-small	PlantVillage (54,303 images)	Accuracy = 99.5%
[11]	VGG16 (Transfer Learning)	Kaggle Rice Leaf Disease (4,500 images)	Accuracy = 90%
[12]	SVM	PlantVillage (2,152 images)	Accuracy = 98.9%
[13]	SVM + GLCM + k-means	PlantVillage (8,350 images)	Precision = 99%
[14]	SVM + GLCM + LBP + k-means	500+ images (4 disease classes)	Accuracy = 97.2%
[15]	KNN + GLCM + k-means	Carrot leaves (undisclosed, est. ~400)	Accuracy = 95%
[16]	CAE + DenseNet121	PlantVillage (9,920 tomato images)	Accuracy = 98.35%
[17]	Deep KNN (optimized)	Multispectral crop leaf dataset	Accuracy = 89.12%

Khan et al. [10] implemented a MobileNetV3-small model on the PlantVillage dataset consisting of over 54,000 images, achieving an accuracy of 99.5%. Similarly, Jangid [11] utilized transfer learning with VGG16 on a rice leaf disease dataset and reported an accuracy of 90%. Abdu et al. [12] compared support vector machine (SVM) models to CNNs for classifying potato diseases, attaining 98.9% accuracy.

Several studies have integrated traditional feature extraction techniques with machine learning classifiers. Ahmed and Yadav [13] applied SVM in combination with GLCM and k-means clustering on the PlantVillage dataset and achieved 99% precision. Jamjoom et al. [14] enhanced this approach by incorporating local binary patterns (LBP), reaching 97.2% accuracy across four disease types. Similarly, Komala [15] employed a combination of k-means clustering,

GLCM, and KNN on carrot leaf images, with an accuracy of 95%.

Recent advancements have also explored the use of deep learning models for feature extraction. Kumar et al. [16] proposed a hybrid approach combining a convolutional autoencoder (CAE) with DenseNet121 for tomato disease classification, reporting an accuracy of 98.35%. In another approach, Gaikwad and Musande [17] utilized a meta-optimized deep KNN model on a multispectral crop dataset, yielding an accuracy of 89.12%.

Previous studies have shown high accuracy in plant disease detection using deep learning and supervised classification approaches. But the methods used are constrained by several limitations. First, most of the research relies on close-up images of individual leaves [10–16], which are impractical for monitoring large plantation areas. Second, these methods generally require large, annotated datasets for training, which are costly and time-consuming to obtain. Finally, many prior works focus on classification into predefined disease types, rather than unsupervised identification of disease-affected areas at the canopy or plantation scale. To address these shortcomings, the present study introduces an unsupervised clustering framework that operates directly on aerial drone imagery without the need for manual labeling. By comparing deep learning-based feature extraction (CAE) with handcrafted texture features (GLCM), we also investigate approaches that can perform robustly under small-sample constraints, making the method more practical for early disease detection in real-world plantation settings.

III. Data and Methodology

This section explains the data and methods used to explore the main topics of this study. Fig. 1 outlines the step-by-step process followed throughout the research.

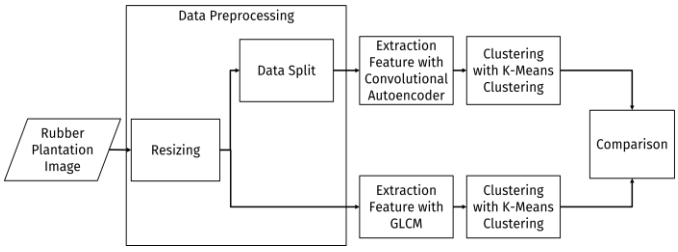


Fig. 1. The research flow of this study.

A. Dataset

The dataset utilized in this study comprises aerial photographs of rubber plantations, collected from the plantation site of PT. Pusat Penelitian Karet Nusantara Sembawa, located in South Sumatra, Indonesia. A total of 19 images were captured using a drone, with each image representing a different section of the plantation and different times, some are taken from 2020 and some are taken from 2021. As illustrated in Fig. 2, the dataset has variation in image perspectives, influenced by differences in drone altitude during image acquisition. Some images were taken from a relatively low altitude, offering a closer view of the canopy (the lower right side of Fig. 2), while others were captured from higher elevations, providing a broader overview of the plantation landscape.



Fig. 2. Sample images of the dataset.

B. Data Preprocessing

- Resizing Data: Resizing images during the data preprocessing stage is a critical step in preparing the dataset for deep learning models. This process involves adjusting the size of images to meet the requirements of the model, with benefits such as computational efficiency, reduced memory load, and ensuring consistency in input sizes for the model, and preventing overfitting. In this study, the image is resized into 20×20 for the input of GLCM and 64×64 for the input of CAE.
- Convert to greyscale: Before applying the Gray Level Co-occurrence Matrix (GLCM) method, the input image must be converted to greyscale. This is because GLCM is designed to analyze texture by evaluating the spatial relationship of pixel intensities, which are represented in a single channel. Color images contain three channels (Red, Green, and Blue), which would complicate the computation. By converting the image to greyscale, it reduces the data to one intensity channel, ensuring that GLCM can accurately process and extract texture features.
- Data Train and Test Split: The division of data into training and test sets is a key stage in the development of models. The training data, which comprises the majority of the dataset, is used to train the model and enable it to understand patterns and features within the data. Meanwhile, test data is employed to assess the model's ability against data it has not previously encountered. In this study, the data was divided into train, validation, and test sets with proportions of 70% and 30%, respectively.

C. Building Models

1) Autoencoder: An autoencoder is a specific type of Artificial Neural Network (ANN) designed to reconstruct its input. Typically, the network's hidden layers engage in

dimensionality reduction on the input, learning prominent features that facilitate effective reconstruction [18]. The main idea of the autoencoder is to have the same input size as the output size. Bottleneck layer is then introduced to make the model able to learn the representation of the data. Autoencoder is consist of four components, that is the bottleneck layer, the decoder network, the encoder network and the reconstruction loss [8].

The encoder network is part of the autoencoder that is used to encode input data into a representation of input data. The mathematical equation that represents each layer at the encoder is presented in Eq. (1):

$$X_{e_{i+1}} = f_{e_i}\left(W_{e_i}^T X_{e_i} + b_{e_i}\right) \forall i = 0,1,2,\ldots,N. \quad (1)$$

The mathematical notation used in the encoder network is X_{e_i}, $X_{e_{i+1}}$, W_{e_i}, b_{e_i}, $and\ f_{e_i}$, which are the input of i^{th} layer, the output of the i^{th} layer, the weight of the i^{th} layer, bias of the i^{th} layer, and the activation function of the i^{th} layer, respectively. The encoder network's final layer is called the bottleneck layer, and its output is a representation of the input data. As a component of the autoencoder, the decoder layer uses the bottleneck layer's output as input and attempts to remodel input data from its representation. The mathematical equation that represents each layer at the decoder is presented in Eq. (2):

$$X_{di+1} = f_{di}(W_{di}^T X_{di} + b_{di}) \forall i = 0,1,2,\ldots,N \quad (2)$$

The mathematical notation in the decoder network is $X_{d_i}, X_{d_{i+1}}, W_{d_i}, b_{d_i}$,and f_{d_i}, which are the input of i^{th} layer, the output of the i^{th} layer, the weight of the i^{th}layer, bias of the i^{th} layer, and the activation function of the i^{th} layer, respectively.

Reconstruction loss is the difference between the original data X^O and the remodelled data X^R. Autoencoder is trained to minimize reconstruction loss. There are two reconstruction losses commonly used, namely binary cross-entropy (BCE) and mean squared error (MSE). The mathematical equation for mean squared error is shown by Eq. (3). Variable D is the number of samples used.

$$MSE(X^O, X^R) = \frac{1}{D}\sum_{j=1}^{D}\left(X_j^O - X_j^R\right)^2 \quad (3)$$

A convolutional autoencoder operates based on the same principles as a traditional autoencoder, with the key distinction being the utilization of convolutional layers instead of dense layers [19]. Similar to the autoencoder, the CAE also consist of four parts: the encoder, the decoder, the bottleneck layer and the reconstruction loss. While the types layer used by CAE is similar to the convolutional neural network (CNN), it primarily uses a convolutional layer and a pooling layer in the encoder and a convolutional layer and an upsampling layer in the decoder. The type of layer used by CAE makes it more suitable for the image data. As shown in [20], the convolutional layer performance is better when handling the images compared to the fully connected layer due to the number of parameters used in the convolutional layer being less than fully connected layer, making the convolutional layer harder to overfitting and translational invariance. Layers used in the encoder network primarily aims to extract prominent characteristic, while the layers used in the decoder network primarily aims to reconstruct the compressed input image.

Convolutional autoencoders (CAEs) were selected in this study due to their strong capability for unsupervised feature extraction, especially in settings, where labeled data is scarce. Since CAEs are trained to reconstruct input images, they require no manual annotations, the image itself serves as the training signal. By using convolutional layers, CAEs are able to capture local spatial patterns such as edges, textures, and shapes at multiple scales, which are well-suited for analyzing structural variations in aerial plantation images.

The bottleneck layer compresses the input into a low-dimensional latent space, forcing the network to retain only the most essential information. This enables CAEs to act as powerful non-linear dimensionality reduction tools, outperforming traditional linear methods like PCA in capturing complex feature combinations with higher discriminative power [21].

Additionally, by eliminating redundant or noisy pixel information, CAEs help reduce the risk of overfitting, which is particularly advantageous when training on small datasets such as the 19 aerial images used in this research.

2) Gray level co-occurrence matrix: The texture of a gray image is elucidated through the examination of spatial correlation characteristics of gray levels using the gray level co-occurrence matrix [22]. GLCM analyzes textures in images based on grayscale intensity, distance, and angle between representations of two neighboring pixels [23]. Texture features play a crucial role in distinguishing regions within images. They pertain to visual patterns characterized by homogeneity, avoiding the production of a single color or intensity [22]. The gray intensity used in digital image data has a value range of 0 to 255 or a quantization level of 256. The quantization level is the number of possible values that an image will have. Some of the texture metrics that can be used are dissimilarity, contrast, homogeneity, energy, angular second moment (ASM), and correlation.

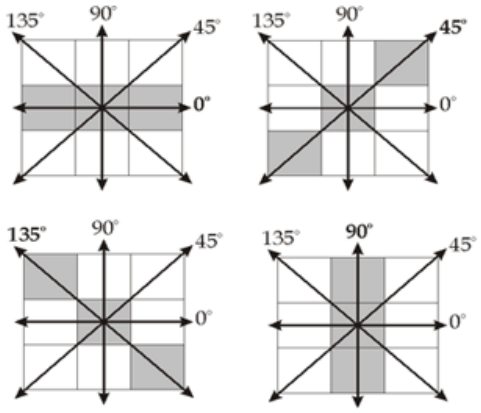


Fig. 3. Angle variation in GLCM.

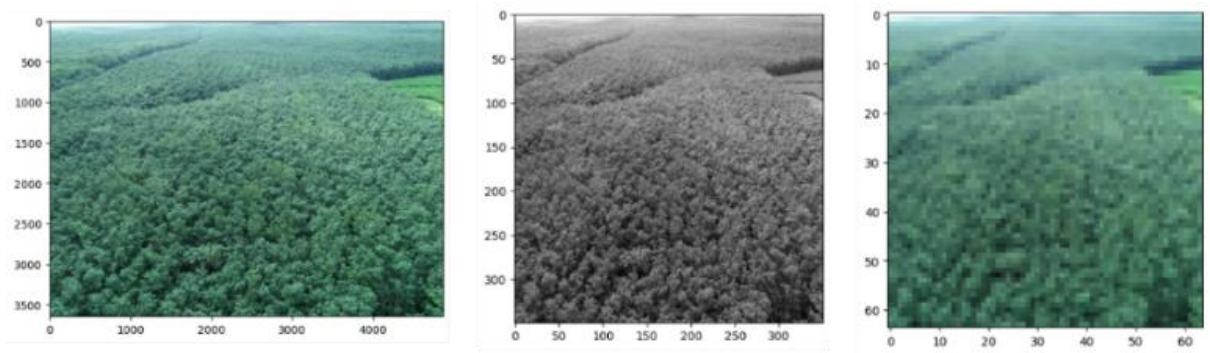
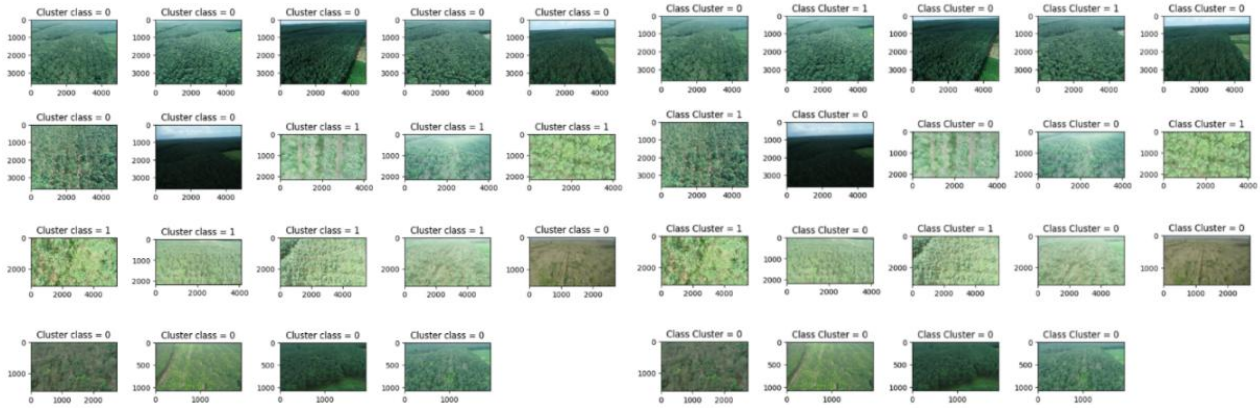


Fig. 4. Comparison between original image (left) and resized image (middle and right).



(a) (b)

Fig. 5. Clustering result for CAE combined with k-means clustering (a), Clustering results of GLCM combined with k-means clustering (b).

Contrast is the magnitude of the difference in color intensity at a pixel coordinate point (x, y) in the image with the surrounding pixels. Dissimilarity is a feature that states the size of the variation in pixel values in the image. Homogeneity is a feature that measures the distribution of each entry value in the GLCM matrix with its diagonal entries. ASM is a measurement of the similarity of a picture. A similar picture will only carry a few gray values; therefore, GLCM will have a slight variation in the value of $(i,)$, but each (i,j) will have a high value. Energy is the square root of ASM, which has a value range from 0 to 1, with 1 representing a constant image or an image whose entry value is the same for each pixel coordinate. Correlation is a measurement of how correlated a pixel coordinate value is with the surrounding pixels. In the GLCM method, there are about eight variations of angles that can be used, namely angles 0°, 45°, 90°, 135°, 180°, 225°, 270°, and 315°, as shown in Fig. 3.

The GLCM is a widely used texture-based feature extraction method that captures spatial relationships between pixels at defined directions and distances. It generates a set of statistical features—such as contrast, correlation, energy, and homogeneity—that quantify texture patterns within an image. These features are particularly effective for differentiating image regions based on their structural properties, making GLCM well-suited for classification and segmentation tasks. Moreover, GLCM is inherently color-independent, allowing it to work efficiently with grayscale images. This makes it a highly appropriate choice for analyzing drone-captured plantation imagery, where subtle variations in canopy texture caused by disease can be more informative than color. In agricultural applications, GLCM has been proven to reliably characterize vegetation patterns and support decision-making in crop monitoring.

3) K-means clustering: K-means clustering is among the clustering approaches in unsupervised learning. This method requires the user to determine the number of clusters, and then the dataset will be divided into clusters. The distance between each centroid point in each cluster and the data points is utilized by k-means clustering to sort out the data. The cluster whose centroid is closest to a given data point comprises the data points. In the k-means clustering method, each data point can only be grouped into one cluster or also known as hard clustering. The following is the algorithm of the k-means clustering according to [24].

(1) Determine the number of clusters (K).

(2) Selects K random points as the centroid (μ_k) per cluster.

(3) Label each data based on the nearest centroid point using the Euclidean distance using Eq. (4), where r_{nk} is the label of each x_n that belongs to cluster K.

$$r_{nk} = \begin{cases} 1, if\ k = \min\limits_{j} \left|\left|x_n - \mu_j\right|\right|^2 \\ 0, \ \text{else} \end{cases} \quad (4)$$

(4) Update the centroid point using Eq. (5) based on the cluster formed by Eq. (4).

$$\mu_k = \frac{\sum_{n=1}^{N}(r_{nk}x_n)}{\sum_{n=1}^{N} r_{nk}} \quad (5)$$

(5) Repeat Steps 3 and 4 until the convergence criteria are fulfilled.

D. Experimental Design

This research started by collecting the dataset, which is a rubber plantation image photographed using an aerial drone. The image was then resized into 20×20 and converted into grayscale as GLCM input, while the image resized into $64 \times 64 \times 3$ as CAE input. The dataset will be splitted into train and test dataset for feature extraction using CAE. The image will then be feature extracted with the respective model. Both outputs of the feature extraction model are then clustered using k-means clustering. The clustering result will be evaluated using the silhouette score and davies-boudin index.

The architecture of the convolutional autoencoder (CAE) in this study was designed with a focus on balancing feature richness and overfitting risk, especially given the small dataset (19 images). We used a series of convolutional layers followed by pooling layers in the encoder to progressively reduce spatial dimensions and capture hierarchical patterns in the image. A kernel size of 3×3 was selected to capture fine-grained texture details typical of leaf canopy variation. Pooling layers were used to reduce spatial dimensions while preserving the most salient features. The bottleneck layer was set to 8×8×8 to compress the image while still retaining enough structural information for effective reconstruction and clustering. The decoder mirrored the encoder using upsampling layers to reconstruct the image.

The selection of four principal angles in GLCM 0°, 45°, 90°, and 135°, was made to adequately capture texture orientation diversity across the image. These directions represent the primary spatial alignments typically found in natural and man-made textures, including horizontal, vertical, and diagonal structures. Using these four directions allows the model to extract comprehensive spatial patterns without significantly increasing computational complexity. The features derived from the GLCM—such as contrast, energy, homogeneity, and correlation—statistically represent image texture by quantifying pixel relationships within a local spatial context. These features offer strong descriptive and discriminative power, making them highly effective for a wide range of image analysis tasks, including classification, segmentation, and object detection. In the context of this study, these features help differentiate between dense (healthy) and sparse (infected) canopy textures in aerial plantation imagery.

In this research, the author used two different methods to extract features from photos of rubber plant fields, namely GLCM and a convolutional autoencoder. In the convolutional autoencoder, the data will initially be resized to a size of $64 \times 64 \times 3$. Then the data will be split into training and testing, by testing as much as 30% of the dataset. The CAE architecture used can be seen in Table II. The optimizer used is Adam with a loss function of mean squared error and 200 epochs. In GLCM, first, the data will be resized to a size of 20×20. Then the co-occurrence matrix will be formed. Next, feature extraction will be carried out on the co-occurrence matrix that has been created. After going through the feature extraction, the dataset will be clustered using k-means clustering. For the k-means clustering, the initialization method used is k-means++ to improve and faster convergence. Each experiment was run with 10 different centroid seeds with the maximum number of iterations for each experiment was set to 300. These settings are chosen based on experiment and shows consistent clustering performance and reduced sensitivity to initialization randomness. The results of clustering will be evaluated with a silhouette score and davies-bouldin index.

TABLE II ARCHITECTURE OF CAE

Layer	Layer type	Input shape	Size of kernel	Activation function	Padding	Output shape
1	Input layer	64 × 64 × 3	-	-	-	64 × 64 × 3
2	Convolutional layer	64 × 64 × 3	3 × 3	ReLU	Same	64 × 64 × 16
3	Average Pooling layer	64 × 64 × 16	2 × 2	-	-	32 × 32 × 16
4	Convolutional layer	32 × 32 × 16	3 × 3	ReLU	Same	32 × 32 × 8
5	Average Pooling layer	32 × 32 × 8	2 × 2	-	-	16 × 16 × 8
6	Convolutional layer	16 × 16 × 8	3 × 3	ReLU	Same	16 × 16 × 8
7	Average Pooling layer	16 × 16 × 8	2 × 2	-	-	16 × 16 × 8
8	Convolutional layer (Bottleneck layer)	8 × 8 × 8	3 × 3	ReLU	Same	8 × 8 × 8
9	Upsampling layer	8 × 8 × 8	2 × 2	-	-	16 × 16 × 8
10	Convolutional layer	16 × 16 × 8	3 × 3	ReLU	Same	16 × 16 × 8
11	Upsampling layer	16 × 16 × 8	2 × 2	-	-	32 × 32 × 8
12	Convolutional layer	32 × 32 × 8	3 × 3	ReLU	Same	32 × 32 × 8
13	Upsampling layer	32 × 32 × 8	2 × 2	-	-	64 × 64 × 8
14	Convolutional layer	64 × 64 × 8	3 × 3	ReLU	Same	64 × 64 × 3

IV. Experiments and Result

In this section, we will explain the results of our research and at the same time a comprehensive discussion is also given.

A. The Evaluation Parameters

Silhouette score is an approach applied to assess clustering. The assessment is done by finding the distance between data in one cluster and the distance with other data in different clusters [25]. The mathematical equation for the silhouette score is presented by Eq. (6):

$$S(i) = \frac{p(i)-q(i)}{max(p(i)q(i))} \quad (6)$$

The variable $p(i)$ means the average length of data accompanied by every other data in the similar cluster, while the variable $q(i)$ is the average length of data i with every other data in the nearest cluster, $S(i)$ is the silhouette score of i^{th} data. Silhouette score has a range of values from [-1,1]. A value near 1 denotes an excellent clustering because the data point is closer to its centroid means that each data can be grouped into a cluster that has the same characteristics. Value close to 0 indicates overlapping clusters or the distance between the clusters is not significant. Value close to -1 indicates very poor clustering because the data occupies the wrong cluster [26].

Another metric used to measure the clustering performance is the davies-bouldin Index [27]. Davies -bouldin index measures the average similarity between each cluster and its most similar ones. A lower davies-bouldin index indicates better clustering performance [see Eq. (7)]:

$$DBI = \frac{1}{k}\sum_{i=1}^{k} \max_{j\neq 1}\left(\frac{S_i+S_j}{M_{ij}}\right) \quad (7)$$

where, S_i is the average of intra-cluster distance in cluster i and M_{ij} is the centroid distance between cluster i and j.

B. Results

The outcome of the observations in this study will be discussed here. Fig. 4 displays a comparison of the original and resized images. The original image is shown on the left side, while the resized image is shown on the middle and right sides. The original image in the dataset has a varied shape. The GLCM will reduce the image into 20×20 and convert from an RGB image into a grayscale image, while the CAE will reduce the image into 64×64×3. The output feature extraction for 1 sample in GLCM is a 1D array with a shape of 24, while for CAE, the output size for 1 sample is 8×8×8, but k-means clustering is only able to take input with a shape of a 1D array; the output of CAE will be flattened to transform the output size into 1D array.

The clustering result from CAE combined with k-means clustering is shown in Fig. 5. Clustering result for CAE combined with k-means clustering. It can be seen from 19 samples, 7 of the samples are clustered as infected rubber plant plantation. The clustering results for GLCM combined with k-means clustering is also shown in Fig. 5. From 19 samples, 6 of the samples are clustered as infected rubber plant plantation.

The comparison of silhouette score between CAE and GLCM is also shown. The results of each model reveal that greater the silhouette score value, the better the cluster created by the model. This is because the clusters formed have low distances between data in the same cluster while have large distances between data in different clusters indicating that the clusters formed are well separated. This finding is also supported by the davies-bouldin index of GLCM, which is smaller compared to CAE, indicating better separation. From TABLE IIIresults of each model below, performance of GLCM combined with k-means clustering is better compared to CAE combined with k-means clustering. It means that GLCM combined with k-means clustering has better clustering performance which in this case is separating the healthy rubber plantation and infected rubber plantation.

TABLE III Models Performance

No.	Method	Silhouette Score	Davies-bouldin Index
1	CAE combined with k-means clustering	0.467	1.638
2	GLCM combined with k-means clustering	0.788	0.653

C. Discussion

The performance difference between the CAE and GLCM methods, reflected in silhouette scores of 0.467 and 0.788, respectively, can be attributed primarily to the limitations of the dataset and the characteristics of each method. Besides, the silhouette scores, clustering quality was further assessed by the davies-bouldin index. The CAE methods achieve 1.638, indicating moderate overlap between clusters and have poor performance on separation, while the GLCM has better performance with the davies-bouldin index of 0.653, indicating well-separated clusters. These findings, based on silhouette score and davies-bouldin index, show that the GLCM provides more discriminative representations resulting in more reliable clustering performance.

One major factor is the small sample size. This study only utilized 19 aerial images, which is far below the typical data requirements for deep neural networks like CAE. As explained by LeCun et al. [28], deep learning models rely on large datasets to learn meaningful representations; otherwise, they risk overfitting and poor generalization. In this case, the CAE likely failed to learn robust feature embeddings due to insufficient training data.

Moreover, the curse of dimensionality further exacerbated the CAE's performance. The output feature map of size 8×8×8 was flattened into a 1D vector of 512 features per image, resulting in a dataset of shape (19, 512). With far more features than samples, the model operated in a highly sparse input space, where distances between points become less meaningful, making clustering less effective [29].

In contrast, GLCM produces a smaller set of statistical texture features that are handcrafted and less sensitive to data volume. These features capture relevant spatial patterns in grayscale images without the need for extensive training. As a result, the GLCM-based clustering method was more robust

and better suited to the constraints of a small dataset. This performance gap illustrates the importance of aligning model complexity with data availability: simpler feature extraction techniques may outperform deep learning approaches in low-data scenarios.

This study acknowledges some limitations. The dataset used in this work consists of only 19 images collected from a single plantation, without ground-truth labels or cross-site validation. The clustering results, silhouette score, and davies-bouldin index may be influenced by dataset-specific characteristics rather than genuine disease discrimination. To address this, future research will focus on expanding the dataset to include more than one plantation site and diverse conditions. Ground-truth labels obtained through expert annotation will be incorporated to enable supervised validation.

V. Conclusion

Clustering with images of a rubber plantation as a dataset can be done with a machine learning approach. First, gather images of a rubber plant plantation. Second is to resize the image. Third, extract prominent features from the image using feature extraction. In this study, the feature extraction methods used are CAE and GLCM. Fourth, clustering the extracted image using k-means clustering. The last step is to evaluate the result using the silhouette score and davies-bouldin index. GLCM combined with k-means clustering have better performance compared to CAE combined with k-means clustering, meaning that GLCM combined with k-means clustering is able to separate the healthy rubber plantation and infected rubber plantation.

This study has several limitations. The most prominent being the small dataset size of only 19 aerial images. Such a limited sample restricts the generalizability of the results and poses challenges for deep learning models like the convolutional autoencoder (CAE), which typically require large datasets to learn robust, non-overfitted representations. The risk of overfitting in the CAE is especially high under these conditions, as the model may memorize noise or irrelevant patterns rather than general features of healthy or diseased plantation areas. Additionally, the dimensionality of the CAE's output (512 features) compared to the number of samples introduces sparsity that negatively affects clustering.

Future works should consider data expansion, such as collecting larger and more diverse aerial datasets across different seasons and regions to ensure generalization. Methodologically, hybrid approaches that combine handcrafted features (for example, from GLCM) and deep learning representations (for example, CNN or CAE) should be explored. Alternative approaches of clustering methods, such as Density-Based Spatial Clustering of Application with Noise (DBSCAN) or Gaussian Mixture Model, may better capture complex data patterns. Finally, integrate the proposed framework into a real-time disease monitoring pipeline.

Acknowledgment

This research is funded by Research and Innovation Program Grants for Advanced Indonesia, number: 36/IV/KS/06/2022 and number: 250./PKS/WRII-DISTP/UI/2022 and supported by Data Science Center (DSC) FMIPA UI, Science and Technology Research Partnership for Sustainable Development (SATREPS) Project, Japan International Cooperation Agency (JICA), RIKEN and Indonesia Rubber Research Institute (IRRI).

References

[1] F. Zuhdi, "The Indonesian natural rubber export competitiveness in global market," Int. J. Agric. Syst., vol. 8, no. 2, pp. 130–139, Jan. 2021.

[2] T. R. Febbiyanti, "Mitigasi kehilangan produksi akibat penyakit gugur daun Pestalotiopsis," in Proc. Konferensi Nasional Karet, 2022.

[3] I. Warsito, "Peluang dan tantangan investasi agribisnis karet alam Indonesia," in Proc. Konferensi Nasional Karet, 2022.

[4] N. Damiri, Y. Pratama, T. R. Febbiyanti, S. E. Rahim, D. T. Astuti, and Y. Purwanti, "Pestalotiopsis sp. infection causes leaf fall disease of new arrivals in several clones of rubber plants," Biodiversitas J. Biol. Divers., vol. 23, no. 8, pp. 3943–3949, Aug. 2022.

[5] G. Sitanggang, D. D. Domiri, I. Carolita, and H. Noviar, "Model spasial indeks luas daun (ILD) padi menggunakan data TM-Landsat untuk prediksi produk padi," Semant. Scholar, Oct. 2010.

[6] P. J. K. Alchemi and S. Jamin, "Impact of Pestalotiopsis leaf fall disease on leaf area index and rubber plant production," IOP Conf. Ser.: Earth Environ. Sci., vol. 995, no. 1, Apr. 2022.

[7] W. K. Mutlag, S. K. Ali, Z. M. Aydam, and B. H. Taher, "Feature extraction methods: A review," J. Phys.: Conf. Ser., vol. 1591, Jul. 2020.

[8] P. Bedi and P. Gole, "Plant disease detection using hybrid model based on convolutional autoencoder and convolutional neural network," Artif. Intell. Agric., vol. 5, pp. 90–101, 2021.

[9] M. Yunus, "Feature extraction: Gray level co-occurrence matrix (GLCM)," Medium, Jul. 2020.

[10] A. T. Khan, S. M. Jensen, A. R. Khan, and S. Li, "Plant disease detection model for edge computing devices," Front. Plant Sci., vol. 14, pp. 181–204, Dec. 2023.

[11] S. K. Upadhyay, R. Singh, A. Kumar, and N. Sagar, "Rice Leaves Disease Detection Mechanism Using VGG16 Deep Learning Architecture" Innovations in Data Analytics (ICIDA 2023), vol. 1005, Sep. 2024.

[12] A. M. Abdu, M. M. Mokji, and U. U. Sheikh, "Machine learning for plant disease detection: An investigative comparison between support vector machine and deep learning," IAES Int. J. Artif. Intell., vol. 9, no. 4, pp. 670–683, Dec. 2020.

[13] I. Ahmed and P. K. Yadav, "A systematic analysis of machine learning and deep learning based approaches for identifying and diagnosing plant diseases," Sustain. Oper. Comput., vol. 4, pp. 96–104, Mar. 2023.

[14] M. Jamjoom, A. Elhadad, H. Abulkasim, and S. Abbas, "Plant leaf diseases classification using improved k-means clustering and SVM algorithm for segmentation," Comput., Mater. & Continua, vol. 76, no. 1, pp. 367–382, Jun. 2023.

[15] T. Komala, S. S. Ashwini, and M. Z. Kurian, "Prediction of plant leaf disease using image pre-processing and filter based optimal feature selection for KNN classifier," Int. J. Adv. Res. Ideas Innov. Technol., vol. 7, no. 3, pp. 510-513, 2021.

[16] T. Kumar, et al., "Plant disease detection and classification using hybrid model based on convolutional auto encoder and convolutional neural network," Comput., Mater. & Continua, vol. 83, no. 3, pp. 5219–5234, 2025.

[17] V. P. Gaikwad and V. Musande, "Advanced prediction of crop diseases using Cetalatran-optimized deep KNN in multispectral imaging," Trait. Signal, vol. 40, no. 3, pp. 1093–1106, Jun. 2023.

[18] M. Maggipinto, C. Masiero, A. Beghi, and G. A. Susto, "A convolutional autoencoder approach for feature extraction in virtual metrology," Procedia Manuf., vol. 17, pp. 126–133, 2018.

[19] K. O'Shea and R. R. Nash, "An introduction to convolutional neural networks," arXiv preprint arXiv:1511.08458, Nov. 2015.

[20] P. Pi and D. Lima, "Gray level co-occurrence matrix and extreme learning machine for Covid-19 diagnosis," Int. J. Cogn. Comput. Eng., vol. 2, pp. 93–103, Jun. 2021.

[21] H. Niu, G. B. McCallum, A. B. Chang, K. Khan, and S. Azam, "Exploring unsupervised feature extraction algorithms: Tackling high dimensionality in small datasets," Sci. Rep., vol. 15, no. 1, pp. 21973, 2025.

[22] H. Halim, J. Hendryli, and D. E. Herwindiati, "Online product search using gray level co-occurrence matrix, color moments, and histogram of oriented gradients for content based image retrieval," IOP Conf. Ser.: Mater. Sci. Eng., vol. 852, Jul. 2020.

[23] K. Trang, L. TonThat, and G. Minh, "Plant leaf disease identification by deep convolutional autoencoder as a feature extraction approach," in Proc. Int. Conf. Electrical Eng./Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), pp. 522-526, Jun. 2020.

[24] C. Bishop, Pattern Recognition and Machine Learning. New York, NY, USA: Springer, 2006.

[25] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," J. Comput. Appl. Math., vol. 20, pp. 53–65, Nov. 1987.

[26] D. L. Davies and D. W. Bouldin, "A cluster separation measure," IEEE Trans. Pattern Anal. Mach. Intell., vol. PAMI-1, no. 2, pp. 224–227, Apr. 1979.

[27] Y. Januzaj, E. Beqiri, and A. Luma, "Determining the optimal number of clusters using silhouette score as a data mining technique," Int. J. Online Biomed. Eng., vol. 19, no. 4, pp. 174–182, Apr. 2023.

[28] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, May 2015.

[29] N. Venkat, "The curse of dimensionality: Inside out," RG Open, 2020.

Tamil Handwritten Character Recognition: A Comprehensive Review of Recent Innovations and Progress

Manoj K, Iyapparaja M
School of Computer Science, Engineering and Information Systems Vellore Institute of Technology, Vellore, Tamil Nadu, India 635810

***Abstract*—Recognizing handwritten characters is a complex task, particularly when dealing with Tamil, a writing system known for its intricate and stylized nature. Several challenges arise in recognizing Tamil handwritten characters, including the complexity of the writing style, similar character shapes, irregular handwriting, slanting characters, varying curves, inconsistent font sizes, and limited datasets. Additionally, the diversity of writing styles and the absence of a standard solution for accurately recognizing all Tamil characters further complicate the process. To address these issues, researchers have explored various techniques, including neural networks, support vector machines, clustering, and groupwise classification. However, Tamil handwritten character recognition remains an evolving field with ample opportunities for exploration and advancement. This review study aims to provide a thorough analysis of the current state of the field, identify key challenges, and highlight areas for improvement. Furthermore, it presents a detailed examination of the proposed techniques and suggests potential directions for future research in this domain.**

***Keywords*—Convolutional Neural Network (CNN); handwritten recognition; Tamil characters; offline recognition; feature extraction techniques; neural network architecture; Support Vector Machine (SVM); groupwise classification**

I. INTRODUCTION

Handwritten recognition is a crucial aspect of computer vision and has various practical applications such as reading handwritten documents, name recognition, postal address recognition and so on [1]. The process of recognizing Tamil handwritten characters has been a topic of interest in the field of computer vision and pattern recognition [2]. Due to the increasing trend of digitalization, there is an escalating need for precise and effective techniques that can identify handwritten characters and convert them into a digital text format [3]. Identifying Tamil characters, especially in handwritten form, is a difficult task as it involves various hurdles, such as slanted characters, similar-looking characters, connected characters, and curved shapes [4]. However, these techniques have their own limitations and may not provide optimal accuracy and efficiency. Multiple methods have been used to tackle the challenges in the recognition of handwritten characters, including groupwise classification, clustering, support vector machine (SVM), and the Component labeling method. However, these methods have limitations in terms of accuracy and efficiency [5].

Recently, Convolutional Neural Network (CNN) has emerged as a promising solution for handwritten recognition as it can effectively classify and recognize characters [6]. Unlike traditional feature extraction techniques, the system being proposed uses the pixel values obtained from the resized characters during the segmentation stage, which results in a less complex system [7]. In recent times, Artificial Neural Network (ANN) has been used for handwritten recognition. This approach has shown promising results in terms of accuracy and speed [8]. The process of ANN for handwritten recognition involves various stages such as segmentation, feature extraction, and classification. The pixel values obtained from the resized characters during the segmentation stage are utilized to train the neural network. The precision of the recognition system is dependent on the quantity of hidden layers and the number of neurons present in each layer [9]. Moreover, the proposed system is designed to recognize a minimal character set and has limitations in terms of different writing styles and font sizes [10]. However, the system can be further improved in the future by exploring the key challenges in this field, as shown in Fig. 1. The proposed system has the potential to find applications in handwritten name recognition, postal address recognition, and document reading. Through this, our aim is to make a contribution to the progress of research in the field of handwritten recognition and bring us a step closer to a standard solution for identifying Tamil characters with a satisfactory degree of accuracy.

This study is structured as follows: In Section I, an introduction is provided. Section II focuses on the motivation behind the study. Section III highlights the summarized literature review. Table I compares and contrasts different studies. Section IV discusses the challenges in handwritten Tamil character recognition. In Section V, the study concludes by summarizing the findings and outcomes of the study.

A. Tamil Handwriting Recognition

Tamil Handwriting recognition is a field of study that aims to develop algorithms and systems capable of recognizing the handwritten Tamil script [11]. With the increasing amount of digital data and the need for efficient data processing [12], the importance of recognizing and processing handwritten data has become increasingly important. The Tamil script is a complex script, with a large number of characters and variations in handwriting styles, making the recognition process challenging [13]. Numerous

algorithms and approaches have been developed over the years to tackle this challenge. These include traditional methods like clustering and groupwise classification, support vector machines (SVM), Component labeling methods, and Artificial Neural Networks (ANN). While these methods have provided some level of accuracy in recognizing Tamil handwriting, they have certain limitations, such as being time-consuming and complex [14]. The architecture for Tamil handwritten character recognition (HCR) typically involves several stages, including preprocessing, segmentation, feature extraction, and recognition. During the preprocessing stage, the input image is transformed into a form that is suitable for processing, such as converting it to grayscale or thresholding it [15]. The segmentation stage is used to separate individual characters from the background. Feature extraction techniques are employed to obtain significant features from the segmented characters that can be used to represent them. The recognition stage is where the extracted features are utilized to classify the characters and identify their identity [16]. In summary, Tamil Handwriting recognition remains a challenging and ongoing field of study, with ongoing research aimed at improving the accuracy and efficiency of recognition algorithms. With advanced techniques such as deep learning and CNNs, there is great potential for further improvements in this field, which has applications in various areas such as document reading, handwritten name recognition, and postal address recognition.

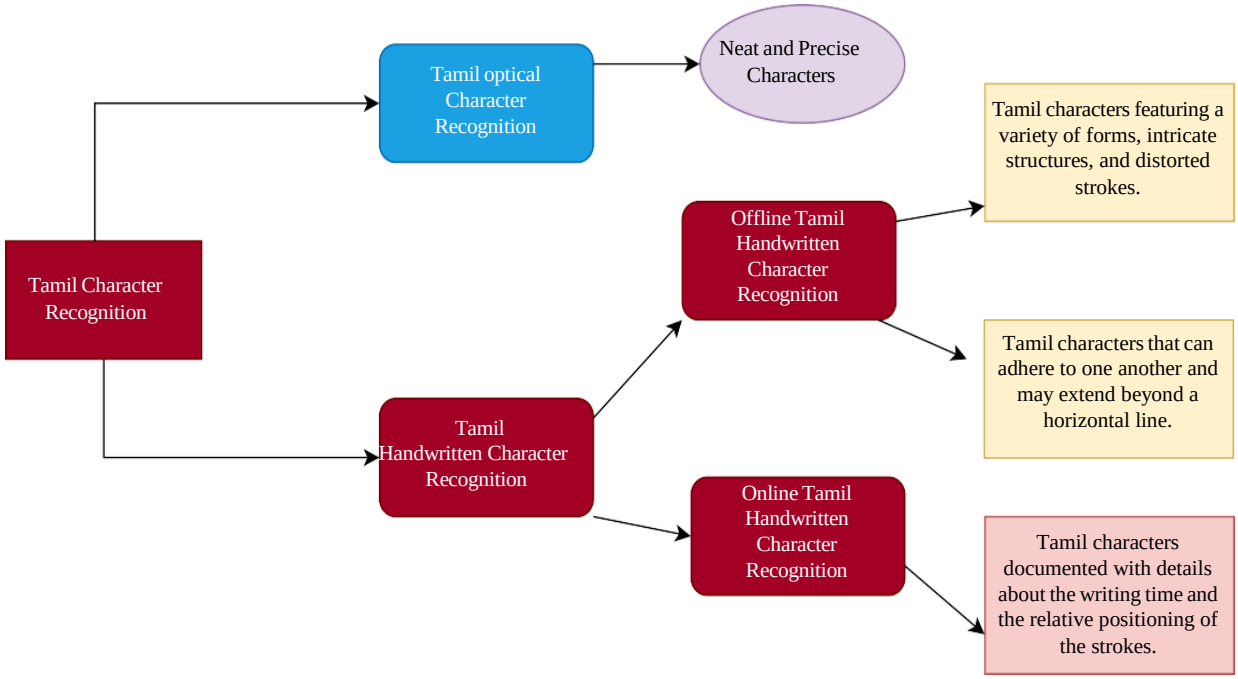


Fig. 1. Classification of Tamil character recognition.

TABLE I COMPARATIVE STUDY

S.No	Methodology	Achievement	Limitations	Ref No
I	Optimized (CNN)	The proposed CNN + SALA model achieves an accuracy of 0.84 at TD=50, outperforming RNN, LA, EHO-NN, and DCNN by 12.7%, 3.4%, 7.2%, and 11.2%, respectively.	The results reveal that although the proposed TCR system performs better than existing models in terms of accuracy, sensitivity, specificity, and precision, it still needs further improvements to achieve optimal recognition of Tamil characters.	[30]
II	MNM (NN+EHO)	Modified traditional neural network that utilizes Elephant Herding Optimization for optimizing weights for the identification of Tamil characters. The MNN method demonstrates an impressive recognition rate of 93%, surpassing the results achieved by other existing methods such as NN, SOM, SVM, FNN, RBF, and Quad tree.	Only Tamil character recognition. Limited comparison to existing classifiers. Evaluation based only on recognition rate.	[19]
III	ResNet (type of ANN with multiple layer skips and nonlinearities, including batch normalization)	Proposed system for Tamil character recognition has 96% accuracy, using a dataset of over 15,000 images of 128x128 dimensions.	Limited to recognition of only 128x128 images of Tamil characters [31]. Similar characters may cause recognition errors. Limited to Tamil character recognition only, cannot recognize words or sentences.	[32]
IV	Support Vector Machines (SVM)	The system has obtained 82.04% recognition accu-racy on handwritten Tamil characters.	Trained with only 35,441 characters from 106 distinct Tamil characters handwritten by 117 individuals. Recognition accuracy varies	[22]

			for different Tamil characters, ranging from 62.84% to 98.9%.	
V	A modified convolutional neural network (M-CNN)	The system achieved a high recognition accuracy of 97.07% for the recognized characters using the proposed method.	The work is limited to isolated characters and there is scope for future work on recognizing Tamil handwritten words and sentences.	[33]
VI	Java Neural Network	Achieved 100% recognition rate with 0.1 second recognition speed and 97% accuracy on the test set.	Only 25 users' handwriting was used for training, with 15 users for testing. Recognition is limited to handwritten Tamil characters from 25 selected users.	[34]
VII	Hidden Markov Models (HMM)	The system's performance for retrieving handwritten documents is 80.75%.	The system has been designed and tested for the Tamil language only. Effectiveness on documents with complex layouts and multiple font styles, as well as handling degraded or damaged images, is limited.	[35]
VIII	SVM + Modified Quadratic Discriminant Function (MQDF)	Evaluated on 36,172 data points, achieving an accuracy of 95.13%.	Limited sample size. Limited feature consideration. Limited script coverage. Limited error analysis. No real-world testing.	[36]
IX	SVM with Feature Selection Algorithms	Achieved 89% accuracy for testing samples.	Optimized for cursive characters, which may result in errors if the writing style varies significantly. Limited feature selection and statistical-based feature extraction algorithms.	[37]
X	SVM, ELM, and MLP Classifiers	The highest accuracy of 95.26% was achieved by the MVTI-SVM combination.	Comparison limited to 4 feature extraction methods. Structural features could enhance recognition accuracy, particularly for ambiguous character pairs.	[38]
XI	ADAM Optimizer	CNN + ADAM achieved 92% accuracy, CNN + Lion Optimization achieved 86%, RMSProp achieved 89%, and AdaGrad achieved 88%.	Processing time is high with the ADAM Optimizer.	[39]
XII	Convolutional Neural Networks	The best neural network architecture for character recognition had two hidden layers of 100 neurons each, with 90.19% accuracy.	Does not cover different writing styles and font sizes. Limited character set. Neural network architecture may not suit other writing styles.	[40]
XIII	Convolutional Neural Networks	A proposed system uses a CNN to recognize 70 Tamil characters with 93% accuracy, outperforming methods like Clustering, SVM, Component labeling, and ANN.	The dataset is limited to 200-250 samples per character from the hp-labs-Tamil-dataset. Only online datasets with specified class labels are supported. Recognizes a limited number of characters.	[41]

II. Motivation

The motivation behind researching handwritten character recognition in Tamil lies in the growing need for fast and accurate recognition of handwritten characters in various domains such as healthcare, transportation, finance, and education. Despite being a complex task, the application of handwritten character recognition in these domains can bring about significant improvements and streamline processes. The challenge arises from the intricate variations in writing styles and individual differences among writers, making traditional recognition methods inadequate. To overcome these difficulties, researchers have been exploring the use of various algorithms for handwritten character recognition. The parallel architecture and fault-tolerance have demonstrated promising results in addressing the intricacies involved in character recognition. The primary objective of this research study is to provide a comprehensive overview of the current state-of-the-art techniques and methodologies used in the domain of Tamil handwritten character recognition. Additionally, this study seeks to identify areas for future research and contribute to the development of advanced and dependable systems for recognizing handwritten characters in Tamil.

III. Summarized Literature Review

Kavitha et al. [17] presents a study on recognizing offline handwritten Tamil characters through the use of Convolutional Neural Networks (CNNs). The study aims to establish a standard for offline Handwritten Tamil Character Recognition (HTCR) by leveraging deep learning techniques. The researchers utilized an isolated handwritten Tamil character dataset that was developed by HP Labs India and trained the model from scratch, achieving good recognition results with a training accuracy of 95.16%. The Tamil Language has a unique feature of having a syllable for every sound pronounced, making the word economy of characters minimal. The Tamil script comprises a total of 307 characters, but the complete Tamil character set can be represented by only 156 distinct characters. This study is significant because it establishes the standard for HTCR for all 156 classes through the application of deep learning methods, unlike previous studies, which used a limited number of classes or used grouping methods.

The field of optical character recognition (OCR) has gained momentum in recent years due to the growing need to convert scanned images into machine-readable formats. The OCR process finds applications in various fields, such as intelligent scanning machines, language translators, and text-to-speech converters. The development of a Tamil character recognition (TCR) system is one of the challenging tasks in OCR, primarily due to the unique problems that the Tamil language poses to the developers. The existing TCR systems face challenges such as variations in the size and style of

handwriting, orientation angle, and font size [18].

To overcome these challenges, Kowsalya et al. [19] proposed an efficient Tamil character recognition system that involves four primary processes: preprocessing, segmentation, feature extraction, and recognition. The preprocessing process includes Gaussian filtering, binarization, and skew detection. The segmentation process involves line and character segmentation. The extracted features are then used for recognition, which is carried out using an optimal artificial neural network. To optimize the weights of the neural network, the researchers used the Elephant Herding Optimization algorithm. The performance of the proposed method was evaluated using sensitivity, accuracy metrics, and specificity. The proposed method is implemented in MATLAB and has shown promising results in the analysis. Previous research has focused on developing efficient TCR systems, and significant progress has indeed been made in the field of Tamil character recognition over the years. However, there are still challenges related to existing character recognition methods, such as the loss of information and change in shape after binarization. The proposed method aims to address these challenges by incorporating effective preprocessing and a modified neural network. The proposed method has the potential to serve as a solution for the development of an accurate Tamil character recognition system. The objective of the paper by Harini et al. [20] is to digitize old handwritten Tamil scripts through the use of Optical Character Recognition (OCR) technology. Tamil is a classical language of India with a rich literary tradition spanning over four thousand years, with much of its literature still undigitized. The OCR system in this study uses four main modules: Scanning, Preprocessing, Recognition, Feature Extraction, and Image Segmentation. The input is a scanned image of handwritten Tamil scripts, which undergoes processes like binarization, noise removal, skew correction, segmentation, and feature extraction to produce a digital output. This module compares the input sample to the trained dataset and generates a digital version of the handwritten input as output.

Raj et al. [21] review pertains to a study on a Tamil handwritten character recognition system. The primary goal of this study is to extract and identify distinct features of different Tamil characters that exhibit variations in their shapes and styles. Three methods of feature extraction were experimented with: shape, location-based instances, and shape ordering. The proposed system consists of four phases: pre-feature extraction, feature extraction, Z-ordering, and classification. The feature extraction phase involves strip tree, Z-ordering, and PM- Quad tree algorithms. The strip tree algorithm deals with shape features, the Z-ordering algorithm is used to address the problem of structure ordering, while the PM-Quad tree is used to facilitate efficient searching and retrieval of characters. The classification phase uses a hierarchical support vector machine to predict characters based on a divide-and-conquer procedure. The researchers collected data for the experiment from the HP-India Handwritten Tamil Character Dataset and individuals of different age groups. They utilized a total of 10,000 characters, with 6,000 characters being used for training and 4,000 characters for testing purposes. The experiment was implemented in MATLAB, and the results show that SVM performed better than other classifiers. The study concludes that the proposed system can recognize a larger number of characters and their varied shapes.

Shanthi et al. [22] aim to recognize offline handwritten Tamil characters using a support vector machine (SVM). The study uses handwritten characters collected from various writers on A4-sized documents, scanned at 300 dpi resolution and stored as grayscale images. Pre-processing is done on the digitized images to improve quality, and pixel densities are computed for 64 different zones to form character features used for SVM training. The system achieves an 82.04% recognition accuracy on the handwritten Tamil character database.

Sornam et al. [23] focused on introducing a new approach to recognize offline handwritten Tamil characters, which is challenging due to their complexity and variability in writing style among different users. Their method involves a combination of principal component analysis (PCA) and a convolutional neural network (CNN) for feature extraction and recognition of Tamil characters, as shown in Fig. 2, resulting in a more accurate recognition system. Deep learning, a state-of-the-art technique in machine learning [24], was used to reduce the burden on the programmer and to extract reliable features for complex problems such as object recognition and handwritten character recognition. The dataset utilized for the research was gathered from diverse writers located in Bangalore and Salem, India. It included both offline and online versions of Tamil characters. The model was trained for 50 epochs and achieved a convergence of approximately 88% in the 27th epoch. The training accuracy increased over the 50 epochs, and the performance measures were promising. The study presented a Tamil character recognition system that showed good performance. The system has the potential to be further extended for recognizing slogans in images with the use of a GPU.

The article by Jose et al. [25] discusses a wavelet transform-based feature extraction technique for recognizing Tamil handwritten characters. The classification process in the algorithm is carried out using a feed-forward back propagation neural network, while feature extraction is done through two-dimensional discrete wavelet transformations. The article highlights the challenges in recognizing Tamil characters, including the large number of characters and complex handwriting styles, and notes that preprocessing, segmentation, feature extraction, and classification are the major steps involved in character recognition. The proposed system has achieved a maximum recognition rate of 89% after preprocessing, wavelet decomposition, and BPNN classification.

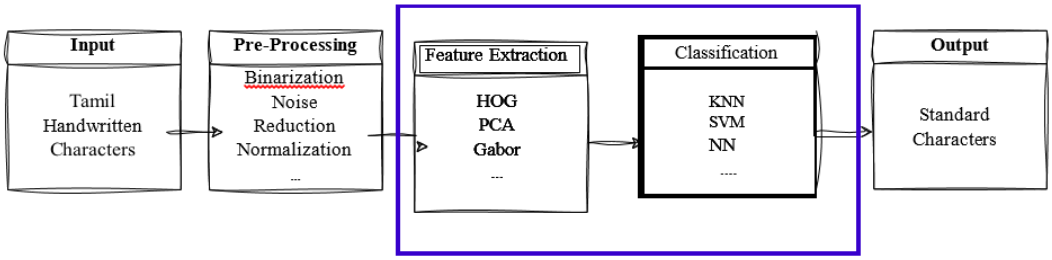


Fig. 2. Traditional offline HTCR process and typical methods for each step.

Wahi et al. [26] concentrate on creating an Optical Character Recognition (OCR) system to recognize handwritten Tamil characters. The study highlights the use of Zernike moments and Legendre polynomials for feature extraction of handwritten Tamil characters and a feed-forward backpropagation neural classifier for character classification. The proposed system consists of three modules: preprocessing, classification and feature extraction. The preprocessing module involves the conversion of color images to binary images through binarization techniques such as thresholding and Otsu's method. The feature extraction module includes the computation of various character properties such as height, width, horizontal lines, vertical lines, slope lines, circles, and arcs. The research concludes that there is a lack of studies on recognizing handwritten characters of Indian scripts, and a complete recognition system is not yet available in India due to the extensive character set of Indian languages, the presence of vowel modifiers and compound characters. The study focuses on the recognition of handwritten South Indian languages, namely Tamil.

Shaffi et al. [27] presents the development of an extensive Tamil Handwritten Character Database (uTHCD) for research in document image analysis. The database consists of 91000 samples in 156 character classes with around 600 samples in each class, collected from 650 volunteers, including school children, teachers, homemakers, faculty and university students. The samples were collected both offline and online and encompass a wide range of writing styles and inherent distortions. The authors anticipate that the database will establish a new standard in research on Tamil handwritten character recognition and provide a starting point for other projects in document image analysis. The study also outlines an experiment conducted on the database using a Convolutional Neural Network (CNN) architecture, which achieved an initial accuracy of 88% on the test data. The existing Tamil Handwritten Character Database, developed by HP Labs India, contains 156 distinct character classes with a varying number of samples in each class. Ad-hoc databases have also been created by various researchers, but the sample size is limited and varies greatly between different works. The uTHCD was created through a sample collection process starting with volunteers writing characters in a 10x8 grid on an A4 sized paper or an electronic form using a digital writing pad. Preprocessing was done to remove unnecessary information and to enhance the quality of the samples. The authors aim to make the database publicly available in raw image and HDF compressed format for easy implementation.

Rajasekar et al. [28] focus on the development of a computer-based handwritten character recognition (HCR) system for Tamil characters. The historical background and development of optical character recognition (OCR) and HCR systems in general are discussed. The article provides an overview of prior handwritten character recognition (HCR) methods and techniques, including those that employ moment-based feature extraction and coding schemes based on neighborhood relations. Data reduction is performed using the fuzzy c-means clustering algorithm, and recognition is carried out using Artificial Neural Networks (ANN). The proposed system in the paper involves three modules: preprocessing, segmentation, and recognition. The preprocessing module includes image filtering and binarization, as well as skew correction. The segmentation module decomposes the image into individual characters. The recognition phase involves the use of correlation and neural network methods. The objective of this study is to create a MATLAB program using the image processing toolbox to develop an OCR system for Tamil alphanumeric and text.

This literature review by Suganthe et al. [29] focuses on the development of a handwritten text recognition (HTR) model for the Tamil language. The study highlights the importance of feature extraction in HTR models and explains how Convolutional Neural Networks (CNNs) can be used for automatic feature extraction. The literature review outlines the steps involved in digitizing handwritten text recognition, including the development of a CNN model, hyper-parameter tuning, character segmentation, and recognition of text by feeding individual characters to the tuned model. The Tamil language is considered as a classical language and has a rich history, with a unique writing system that includes 247 characters, with 156 distinct characters capable of depicting the entire Tamil character set. However, handwritten text recognition for Tamil characters is a challenging task due to various distortions, variations in character length, wide range of handwriting styles, fragmented strokes, and noise. The preprocessing steps involved in the HTR model for Tamil characters include grayscale conversion, binary conversion, foreground character extraction, morphological transformation, line segmentation, and character segmentation. The hp-labs- tamil-iso-character dataset was used in this study and the architecture of the CNN model is discussed in detail, including the use of convolutional and pooling layers. The study concludes with a discussion of the experimental results and future scope of the research.

The proposed literature review by Kishan et al. [42] focuses on the field of Handwritten Character Recognition

(HCR) using Convolutional Neural Networks (CNN). Handwritten character recognition (HCR) is an active and challenging area of research in image processing and pattern recognition, with applications in reading bank cheques and recognizing characters from form applications. The proposed work extends the EMNIST dataset by adding characters from the Tamil language, and trains the CNN using this extended dataset. The feature extraction process includes converting the input image to a grayscale image, normalizing the pixel values, and transforming the input data into a set of features through dimensionality reduction. The Min-Max-Scaler is used to normalize the data, and the CNN is used as a classifier. The system includes preprocessing, feature extraction, Min-Max-Scaler, image normalization, and classification in its architecture. The preprocessing step involves converting the input image to a grayscale image and resizing it to match the resolution of the EMNIST dataset. The process of feature extraction involves transforming the input data into a smaller set of features that can effectively represent the input data. The Min-Max-Scaler normalizes the data to a range between 0 and 1, and the trained CNN is used to recognize the characters and provide the desired output.

Thilagavathi et al. [43] focus on the proposed system for the conversion of handwritten South Indian language characters, specifically Tamil, into digital characters. The learning methodology of the system involves the use of Artificial Neural Networks (ANNs) with the Stochastic Gradient Learning Algorithm and Back-Propagation. Previous implementations of similar systems have shown accuracy rates over 90%. The system architecture includes pre-processing modules for image to pixel conversion, a network trainer, and a character predictor. The dataset used for the system design and implementation is from a worldwide competition on handwritten character recognition. The study also highlights the importance of character recognition in various industries, including banking and subjective answer evaluation, as well as the challenges posed by the Tamil language with its wide use of symbols.

Livingston et al. [44] have developed a handwritten Tamil character recognition system that uses a combination of Hidden Markov Model (HMM) and Fuzzy Logic. The system aims to convert handwritten Tamil characters into Unicode and is a challenging task due to the great variation among handprints. Among the different branches of character recognition, Tamil character recognition is more complicated due to multiple patterns to represent a single character and disconnected multi-stroke characters. The authors propose a framework for handwritten Tamil character recognition using a combination of Hidden Markov Model and Fuzzy Logic. The framework consists of three stages: preprocessing, feature extraction using Hidden Markov Model, and classification using Fuzzy Logic. The Fuzzy technique is used to differentiate between Tamil scripts. The system was tested on 750 samples of Tamil characters and achieved a success rate of 89 to 93%. The authors note that there has been significant research in the field of handwriting recognition, including the use of neural networks and hidden Markov models. While the most popular class of nonlinear vector operators in handwritten recognition is based on order statistics, the authors propose a different approach.

Gayathri et al. [45] conducted research on the recognition of cursive characters in Tamil palm leaf manuscripts using the Convolutional Neural Network (CNN) technique. The authors used a text line slicing segmentation scheme to segment the palm leaf characters, and morphological operations were used for preprocessing to eliminate noise in the input image. The feature processing included text line spacing, spacing without obstacles, and spacing with obstacles. The proposed network achieved 94% accuracy, while the existing ResNet achieved 88% accuracy. The proposed methodology is divided into four steps: i) collection of cursive Tamil palm leaf manuscripts, ii) preprocessing, iii) feature extraction, and iv) classification using CNN. The results and discussion section of the study shows that the proposed CNN model outperforms the current models in terms of accuracy.

The literature review by Suriya et al. [46] focuses on the application of Computational Linguistics and Convolutional Neural Network in the recognition of Tamil characters. The study presents an Intellectual Character Recognition System which uses CNN to recognize the Tamil character dataset and achieved a training accuracy of 99.16%. The challenges in handwritten character recognition are also discussed, including the variations in writing style, difficulties in recognizing similar characters, etc. The study proposed a Character Recognition system using a Convolutional Neural Network and used a dataset developed by HP Labs India, which consists of 156 unique Tamil characters written by native Tamil writers. The study highlights the use of Data Augmentation to improve the variety of data available for training models and overcome the issue of limited data size. The literature review concludes that the proposed system has significant potential in the field of handwritten character recognition, as it is capable of recognizing characters in challenging conditions where traditional character recognition systems fail. The comparative study highlights various methodologies for Tamil handwritten character recognition, with CNN- based approaches achieving the highest accuracy, such as 97.07% using Modified CNNs and 96% with ResNet. Hybrid methods combining CNNs with optimization techniques, like Elephant Herding Optimization, also demonstrated strong performance, achieving up to 93% accuracy. Traditional methods like SVM and HMM showed moderate success, with accuracies ranging from 80.75% to 95.13%, but are limited by smaller datasets and simpler feature extraction techniques. Despite advancements, most methods are restricted to isolated character recognition and rely on small, constrained datasets, limiting generalizability as shown in Table I. The bar chart in Fig. 3 underscores the superior accuracy of CNN-based models, indicating the need for future work on recognizing entire words or sentences and handling diverse datasets.

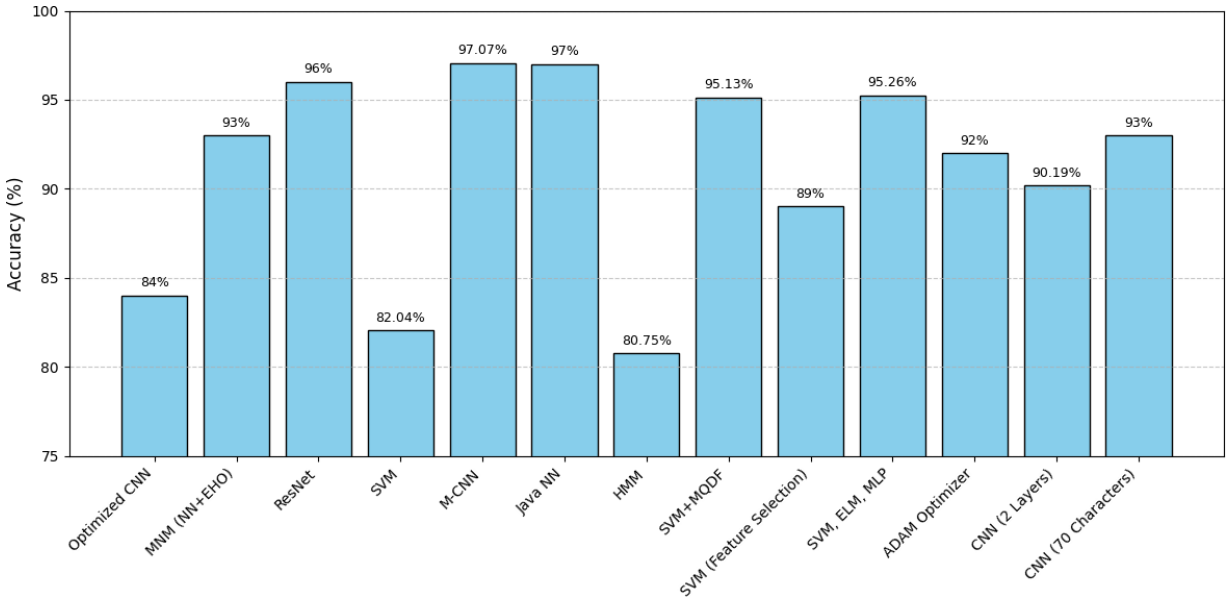


Fig. 3. Comparison of accuracy across methods for Tamil HCR.

IV. Challenges in HCR Tamil

The Tamil writing style can be complex, including variations such as normal and abnormal writing, slanted characters, similar-shaped characters, curves, joined characters, and more, which makes it difficult to accurately recognize the characters [47]. There are many Tamil characters that have similar shapes, which can lead to confusion during recognition [48]. Handwritten characters can be written in an abnormal style, making it challenging to accurately recognize them with standard recognition algorithms [49]. The slanting of characters can also affect recognition, as it can change the character's shape and make it look different from the standard form [50]. Tamil characters can contain curves, which can also pose a challenge for accurate recognition [51]. Different people write in different font sizes, and this variation can pose a challenge for accurate recognition [52]. Feature extraction is often used in recognition processes, but this method can be complex and time-consuming [53]. There is a lack of large datasets for Tamil handwritten character recognition, which limits the development of effective recognition algorithms [54]. People write in different styles, and it can be difficult to accurately recognize characters. In Fig. 4, the Tamil characters have been written in a different style than the one used to train the recognition system [55]. Despite numerous studies and efforts in the field of Tamil handwritten character recognition, there is still a lack of a universally accepted and efficient solution for accurately identifying all Tamil characters [56]. To overcome these challenges, researchers have proposed various approaches including the use of neural networks, support vector machines, clustering, and groupwise classification. The field is still evolving and there is room for further improvement.

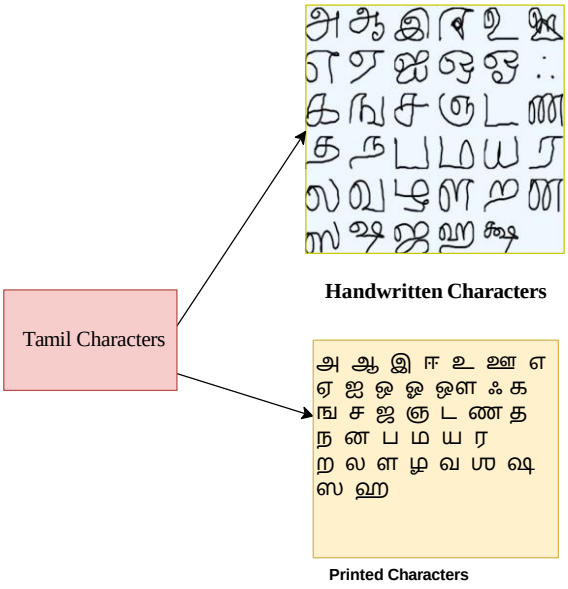


Fig. 4. Comparison of optical vs. handwritten.

V. Conclusion

In conclusion, the research on handwritten character recognition (HCR) in Tamil has made significant strides in recent years. A multitude of algorithms and techniques have been proposed and implemented to tackle the challenges of recognizing handwritten characters in the Tamil script. From the reviewed studies, it can be deduced that Convolutional Neural Networks (CNNs) have emerged as a prominent solution for HCR in Tamil due to their high accuracy and ability to effectively capture the inherent features of handwritten characters. However, there is still room for improvement and further research in this field to enhance the performance and efficiency of HCR systems for Tamil script. This includes the optimization of feature extraction techniques, the development of more robust neural network architectures, and the integration of other advanced machine

learning algorithms. In light of these developments, it is clear that the field of HCR for Tamil script holds immense potential for innovation and advancement, and will continue to play a crucial role in various domains and applications

REFERENCES

[1] Shaveta Dargan, Munish Kumar, Maruthi Rohit Ayyagari, and Gulshan Kumar. A survey of deep learning and its applications: a new paradigm to machine learning. *Archives of Computational Methods in Engineering*, 27:1071–1092, 2020.

[2] B. Radhika. An empirical analysis of Tamil optical character recognition. *Journal of Innovative Image Processing*, 4(2):115– 126, 2022.

[3] S. D. Pande, P. P. Jadhav, R. Joshi, A. D. Sawant, V. Muddebihalkar, S. Rathod, and S. Das. Digitization of handwritten de- vanagari text using cnn transfer learning-a better customer service support. *Neuroscience Informatics*, 2(3):100016–100016, 2022.

[4] K. Subramani and M. Subramaniam. Creation of original Tamil character dataset through segregation of ancient palm leaf manuscripts in medicine. *Expert Systems*, 38(1):12538–12538, 2021.

[5] C Vinotheni and S Lakshmana Pandian. End-to-end deep- learning-based Tamil handwritten document recognition and classification model, 2023.

[6] A. Morera, A. Sa´nchez, J. F. Ve´lez, and A. B. Moreno. Gender and handedness prediction from offline handwriting using convolutional neural networks. *Complexity*, pages 1–14, 2018.

[7] D. S. Prashanth, R. V. K. Mehta, K. Ramana, and V. Bhaskar. Handwritten devanagari character recognition using modified lenet and alexnet convolution neural networks. *Wireless Personal Communications*, 122:349–378, 2022.

[8] J. K. Basu, D. Bhattacharyya, and T. H. Kim. Use of artificial neural network in pattern recognition. *International Journal of Software Engineering and Its Applications*, 4(2), 2010.

[9] M. M. Misgar, F. Mushtaq, S. S. Khurana, and M. Kumar. Recognition of offline handwritten urdu characters using rnn and lstm models. *Multimedia Tools and Applications*, pages 1–24, 2022.

[10] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. volume 39, pages 2298–2304. IEEE, 2016.

[11] E. Vellingiriraj, M. Balamurugan, and P. Balasubramanie. Text analysis and information retrieval of historical tamil ancient documents using machine translation in image zoning. *Inter- national Journal of Language Literature Linguistics*, 2(4):164– 168, 2016.

[12] M. Iyapparaja and P. Sivakumar. Detecting diabetic retinopa- thy exudates in digital image processing hybrid methodology. *Research Journal of Pharmacy and Technology*, 12(1):57–61, 2019.

[13] S. Akashkumar, A. N. Dyaram, and M. Anand. Identification of tamil characters using deep learning. In *Machine Learning and Autonomous Systems: Proceedings of ICMLAS 2021*, pages 223–237. Springer Nature, 2022.

[14] S. Vijayalakshmi, K. R. Kavitha, B. Saravanan, R. Ajaybaskar, and M. Makesh. Handwritten character recognition for tamil language using convolutional recurrent neural network. In *Inventive Systems and Control: Proceedings of ICISC 2022*, pages 369–384. Springer Nature, 2022.

[15] M. A. Mohamad, H. Hassan, D. Nasien, and H. Haron. A review on feature extraction and feature selection for handwritten character recognition. *International Journal of Advanced Computer Science and Applications*, 6(2), 2015.

[16] N. Rani, T. Shobha, M. Vasudev, N. Chandrajith, and Manohar. 2d morphable feature space for handwritten character recognition. *Procedia Computer Science*, 167:2276–2285, 2020.

[17] B Rose Kavitha and C b Srimathi. Benchmarking on offline handwritten tamil character recognition using convolutional neural networks, 2022.

[18] A. K. Sampath and N. Gomathi. Decision tree and deep learning based probabilistic model for character recognition. *Journal of Central South University*, 24:2862–2876, 2017.

[19] S. Kowsalya and P. S. Periasamy. Recognition of tamil handwritten character using modified neural network with aid of elephant herding optimization. *Multimedia Tools and Applications*, 78:25043–25061, 2019.

[20] ML Harini, A Pandian, A Hithayathullah, and AK Sadiq. Handwritten tamil character recognition.

[21] M. A. R. Raj and S. Abirami. Structural representation- based off-line tamil handwritten character recognition. *Soft Computing*, 24:1447–1472, 2020.

[22] N. Shanthi and K. Duraiswamy. A novel svm-based handwritten tamil character recognition system. *Pattern Analysis and Applications*, 13:173–180, 2010.

[23] M. Sornam and C. V. Priya. Deep convolutional neural network for handwritten tamil character recognition using principal component analysis. In *Smart and Innovative Trends in Next Generation Computing Technologies: Third International Conference*, pages 778–787. Springer, 2017.

[24] C. Gopalakrishnan and M. Iyapparaja. Multilevel-thresholding based follicle detection and classification of polycystic ovary syndrome from the ultrasound images using machine learning. *International Journal of System Assurance Engineering and Management*, pages 1–8, 2021.

[25] T. M. Jose and A. Wahi. Recognition of tamil handwritten characters using daubechies wavelet transforms and feed-forward backpropagation network. *International Journal of Computer Applications*, 64(8):975–8887, 2013.

[26] A. Wahi, S. Sundaramurthy, and P. Poovizhi. Handwritten tamil character recognition using zernike moments and legendre polynomial. In *Artificial Intelligence and Evolutionary Algo- rithms in Engineering Systems: Proceedings of ICAEES 2014*, volume 2, pages 595–603. Springer, 2015.

[27] N. Shaffi and F. Hajamohideen. uthcd: a new benchmarking for tamil handwritten ocr. *IEEE Access*, 9:101469–101493, 2021.

[28] M. Rajasekar, A. Bennet, and M. Celine. Performance and analysis of handwritten tamil character recognition using artifi- cial neural networks. *International Journal of Recent Scientific Research*, 7(1):8611–8615, 2016.

[29] R. C. Suganthe, K. Pavithra, N. Shanthi, and R. S. Latha. A cnn model based approach for offline handwritten tamil text recognition system. *NVEO-Natural Volatiles Essential Oils Journal*, pages 164–175, 2021.

[30] R. B. Lincy and R. Gayathri. Optimally configured convolu- tional neural network for tamil handwritten character recogni- tion by improved lion optimization model. *Multimedia Tools and Applications*, 80:5917–5943, 2021.

[31] N. Sankarram B. S. Charulatha R. Jayakanthan, A. H. Kumar and A. Ramesh. Handwritten tamil character recognition using resnet. *International Journal of Research in Engineering*, 3(3):133–137, 2020.

[32] C. Gopalakrishnan and M. Iyapparaja. Detection of polycystic ovary syndrome from ultrasound images using sift descriptors. *Bonfring International Journal of Software Engineering and Soft Computing*, 9(2):26–30, 2019.

[33] S. L. Pandian C. Vinotheni and G. Lakshmi. Modified con- volutional neural network of tamil character recognition. In *Advances in Distributed Computing and Machine Learning: Proceedings of ICADCML 2020*, pages 469–480. Springer, 2020.

[34] C. Sureshkumar and T. Ravichandran. Handwritten tamil character recognition and conversion using neural network. *International Journal on Computer Science and Engineering*, 2(07):2261–2267, 2010.

[35] S. Palanivel A. N. Sigappi and V. Ramalingam. Handwritten document retrieval system for tamil language. *International Journal of Computer Application*, pages 31–31, 2011.

[36] P. Ka and P. Elangob. Accuracy improvement of off-line handwritten tamil character recognition. *Malaya Journal of Matematik (MJM)*, (2):504–512, 2015.

[37] M. A. R. Raj and S. Abirami. Offline tamil handwritten character recognition using statistical features. *Advances in Natural and Applied Sciences*, (9):367–375, 2015.

[38] S. S. Mari and G. Raju. Modified view-based approaches for

handwritten tamil character recognition. *ICTACT Journal on Image Video Processing*, 6(1), 2015.

[39] P. Jagathratchagan R. Iyswarya, S. Deepak and J. Kailash. Handwritten tamil character recognition using convolution neu- ral network by adam optimizer. *International Journal of Advanced Research in Science and Communication Technology (IJARSCT)*, 6(1), 2021.

[40] R. Meena M. Deepa, R. Deepa and R. Nandhini. Tamil hand- written text recognition using convolutional neural networks. *International Journal of Engineering Science and Computing*, 9(3):20 986–20 988, 2019.

[41] A. Sivamurugan S. Ramkumar, M. Saivignesh and K. Shan- mugam. Digitalization of tamil handwritten characters recog- nition using convolutional neural networks (cnn). *International Journal of Advance Research and Innovative Ideas in Educa- tion*, 6:262–268, 2020.

[42] S Anandh Kishan, J Clinton David, P Sharon Femi, and Sri Venkateswara. Handwritten character recognition using cnn. *International Journal of Research and Analytical Reviews*, 5(3), 2018.

[43] G. Lavanya G. Thilagavathi and N. K. Karthikeyan. Tamil handwritten character recognition using artificial neural net- work. *International Journal of Scientific Technology Research*, 8, 2019.

[44] L. G. X. Livingston and L. M. M. Livingston. Handwritten recognition of tamil scripts using hmm with fuzzy logic. *International Journal of Scientific Research*, 2017.

[45] Y. Teekaraman R. Kuppusamy S. G. Devi, S. Vairavasundaram and A. Radhakrishnan. A deep learning approach for recog- nizing the cursive tamil characters in palm leaf manuscripts. *Computational Intelligence And Neuroscience*, 2022.

[46] S. Dhivya S. Suriya and M. Balaji. Intelligent character recognition system using convolutional neural network. *EAI Endorsed Transactions on Cloud Systems*, 6, 2019.

[47] KAKND Dharmapala, WPMV Wijesooriya, CP Chandrasekara, UKAU Rathnapriya, and L Ranathunga. Sinhala handwriting recognition mechanism using zone based feature extraction, 2015.

[48] V. P. Singh H. Singh, R. K. Sharma and M. Kumar. Recognition of online handwritten gurmukhi characters using recurrent neural network classifier. *Soft Computing*, 25:6329–6338, 2021.

[49] A. U. Rahman and Z. Halim. A graph-based solution for writer identification from handwritten text. *Knowledge and Information Systems*, 64(6):1501–1523, 2022.

[50] P. Badelia S. Ghosh A. Chaudhuri, K. Mandaviya. Handwritten text recognition methods and techniques. In *Applications of Pattern Recognition*, pages 9–41. 2017.

[51] T. Deselaers H. A. Rowley A. Daryin M. Calvo Gervais V. Car- bune, P. Gonnet and P. Fast multi-language lstm-based online handwriting recognition. *International Journal on Document Analysis and Recognition (IJDAR)*, 23(2):89–102, 2020.

[52] X. He S. Long and C. Yao. Scene text detection and recogni- tion: The deep learning era. *International Journal of Computer Vision*, 129:161–184, 2021.

[53] L. Fernandes S. Santos M. Abreu P. Bota M. Barandas, D. Fol- gado and H. Gamboa. Tsfel: Time series feature extraction library. *SoftwareX*, page 100456, 2020.

[54] K Shanmugam and B Vanathi. Humming bird optimization based deep belief neural network for tamil handwritten charac- ter recognition, 2024.

[55] R. D. Zarro and M. A. Anwer. Recognition-based online kurdish character recognition using hidden markov model and harmony search. *Engineering Science and Technology, an International Journal*, 20(2):783–794, 2017.

[56] A. Bharath and S. Madhvanath. Handwritten document recog- nition: Challenges and advancements. In *Handbook of Pattern Recognition and Computer Vision*, pages 209–234. 2010.

Boosting Deepfake Detection Accuracy with Unsharp Masking and EfficientNet Models

Radwa Khaled[1]*, Hossam M. Moftah[2], Fahad Kamal Alsheref[3],
Adel Saad Assiri[4], Kamel Hussein Rahouma[5], Mohammed Kayed[6]

Computer Science Department-Faculty of Computer Science, Nahda University, Beni-Suef 62511, Egypt[1]
Faculty of Computers and Artificial Intelligence, Beni-Suef University, Beni-Suef, Egypt[2, 6]
Department of Informatics for Business-College of Business, King Khalid University, Abha 61421, Saudi Arabia[3, 4]
Electrical Engineering Department-Faculty of Engineering, Minia University, Minya 2431436, Egypt[5]

***Abstract*—The rapid progress of deepfake technology, fueled by generative adversarial networks (GANs), has increased the challenge of verifying the authenticity of digital media. This study suggests a more powerful deepfake detection framework based on the EfficientNet convolutional neural network family, coupled with an unsharp masking preprocessing method to highlight manipulation artifacts. Based on a big, diverse dataset of over 5000 video samples, the model was trained and tested on several variants of EfficientNets (B0–B4). The results indicate that the integration of unsharp masking significantly improves the model's ability to detect minor irregularities in facial regions, with its best validation accuracy at 97.77% with EfficientNetB4. The method strikes a balance between computational cost and detection accuracy, rendering it applicable to real-world use cases, such as forensic examination and digital content authentication. The stability of the framework across different datasets and manipulation methods highlights its value as a scalable solution for curbing disinformation and protecting media integrity.**

***Keywords*—Deepfake detection; efficientnet; unsharp masking; convolutional neural networks (CNNs); facial manipulation detection; computer vision; artificial intelligence**

I. Introduction

Deepfakes are photos or videos that have been changed using sophisticated techniques derived from computer vision and deep learning, commonly used to confuse people. In this case, the above methods involve combining, mixing, displaying or exchanging facial attributes to produce artificially authentic albeit synthetic photographs and videos [1].

The act of falsifying data is now widespread in our digital world due to AI and machine learning advancements. The rise in the number of fake images and videos on social media platforms has made reality distortion a significant issue too. Deepfakes are becoming increasingly sophisticated, making it difficult to distinguish them from real content for human beings and harder to catch up with [2].

Deepfake technology is one of the most impactful innovations of manipulating images, audio, or videos, which has made the distinction between real and fake almost incomprehensible. Deepfake videos have gained notoriety primarily because they lead to the deception and manipulation of the audience. These videos utilize AI algorithms to substitute, or overlay faces in the visual footage. The sphere of influence that deep fake videos possess is divergent and multifaceted, and poses significant risks to the media landscape. Such threats range from the distribution of false information to tarnishing a person's reputation or instigating violence. With deepfakes being more common now these days, the need for finding ways to detect and prevent them is becoming more and more crucial [3].

Deepfake is a combination of the two words "Deep learning" and "Fake", and this technology falls on the ends of modifying the video content alongside the Deep Learning (DL) algorithms [4][5]. With the use of deepfake technology, the production of videos and images in an artificial manner has become incredibly easy due to the use of learning networks such as DNNs. One method that can be used to manipulate a video is using a Generative Adversarial Network (GAN) [6], and with it, a single person's image or video can be inserted into another person's content by replacing it with their image and video.

The accomplishment of replacing target faces while maintaining the original voice from the source during the video generation is termed face swapping. This swapping is done on GAN's [7] targets. Zooming in on merging, StyleGAN2 [8] and StyleGAN [9] deepens the layering, which helps to further conceal the images, making it harder for humans to tell the difference between the two images. Initial research [1] [10] indicated that deepfakes were quite discernible, but swift technological development has now made them almost impossible to distinguish from authentic content. The generation of indecent videos using the faces of politicians and celebrities is what has sparked the carrying out of fake news. While this was once an issue raised in society, it is now becoming rampant and easier to get away with, because of the distortion of trust. The shattered trust gives rise to several key social problems, such as the change of public opinion or the spine spreading of fake news [11].

Facial video manipulation technology has evolved to that point in recent years, where humans cannot tell if the video has been manipulated or not [12]. It became a great challenge, especially with deep-fake videos. The deepfake algorithms can manipulate a video in real-time by pasting one person's face onto another or changing the lip movements and facial expressions to make them appear to say anything. Techniques like FaceSwap, Face2Face, NeuralTextures, Deepfakes, and

*Corresponding Author.

face reenactment can create completely new videos that feature the target.

Bad visual quality, unnatural contextual surroundings, or an explicit declaration of being artificial identify some videos as deepfakes. However, several other factors make them very realistic, where the distinguishing factor from actual videos cannot easily be told [13]. Therefore, the detection of a deepfake video has been framed as a binary classification problem, where each image or video is labeled as "real" or "fake".

Numerous techniques have been proposed for deepfake detection; however, they often prove ineffective when applied to real-world videos. External factors such as lighting conditions, compression, scaling, and positional changes further complicate the detection process, making it exceptionally challenging even for advanced deep learning algorithms [11].

Until the year 2006, only convolutional networks and related techniques could be well trained [14]. For the first time, this breakthrough happened in 2006 with the publication of DBN. As mentioned, for the first time in that same year, the term "deep learning" was used by [15].

Neural networks have evolved a lot since then, as the very first ones. Though they may appear advanced and almost futuristic, transforming the notion of a neural network into an actual model that can solve some problem often takes quite a while. It includes data collection, very often with labeling, data preprocessing, and algorithm development. Once the algorithm has been developed, training needs to take place, which is not always easy and may require several runs for optimal performance. Furthermore, a network performing well on a training dataset may not generalize well to unseen data, which is another challenge altogether [16].

A CNN is a particular type of neural network in machine learning applied to medical image analysis. CNNs are designed to process data arrays, like images. The architecture of the CNN consists of three major parts: the input, which is the image in this case, feature extraction, and a nonlinear activation unit. In this context, the kernel can be visualized as a small 2-D matrix that helps to establish relationships between the central pixel and its surrounding pixels, enabling the network to capture spatial patterns effectively [17].

In this work, we present a novel approach to deepfake detection that builds upon and enhances Aaron Chong's implementation [18]. We use an EfficientNet-based model (B0, B1, B2, B3, and B4) and introduce a number of enhancements, including a new sharpening step and utilization of a large, heterogeneous dataset. Our contributions are threefold:

1) Image sharpening step: We introduce an unsharp masking technique to the preprocessing stage, which will serve to improve the detectability of subtle artifacts in manipulated facial regions. The technique, using Gaussian blur and adding the original image, improves high-frequency information, improving artifact detection in deepfakes.

2) Large dataset: We train over a highly diverse and enormous pool of deepfake datasets like DeepFake-TIMIT, FaceForensics++, Google Deep Fake Detection (DFD), Celeb-DF, and Facebook Deepfake Detection Challenge (DFDC) comprising over 134,000 videos. The diversity helps our model to generalize nicely to other kinds of manipulations.

3) Efficient model design: Our network is rooted in the EfficientNet (B0, B1, B2, B3, and B4) models and further optimized through the addition of additional layers for global max pooling and dense fully connected layers. This final design has high accuracy and efficiency for binary classification tasks in predicting the authenticity of videos.

4) Research problem: Recent deepfake detection models are not generalizable across datasets and methods of manipulation. They remain very sensitive to variations in compression, illumination, and facial orientations. This makes them not easily deployable in the real world, where attack vectors keep changing with more sophisticated generation methods.

5) Research objectives: This research aims to address these challenges by:

a) Introducing an unsharp masking preprocessing operation to enhance weak manipulation artifacts in facial regions.

b) Training EfficientNet architectures (B0–B4) on a large and diverse dataset to improve robustness to various manipulations.

c) Creating an efficient and scalable detection framework that sacrifices high classification accuracy for computational efficiency.

6) Research significance: The significance of this study lies in its scientific and practical contributions. Academically, it takes the state of deepfake detection to the next level by combining novel techniques in preprocessing with efficient deep learning models. Practically, the proposed framework benefits digital forensics, strengthens media integrity, and provides scalable solutions for combating misinformation in resource-constrained environments such as social media platforms and police investigations.

This study robusts deepfake detection by coupling a novel preprocessing boost with an advanced model structure for improving both detection accuracy and generalizability. The rest of the study follows the structure as follows: Section II outlines related works. Section III presents the proposed pipeline. Section IV presents experimental results. Section V provides results. Section VI presents the discussion, and Section VII details the ethical concerns, data protection and misuse threat. Section VIII concludes the study. Finally, Section IX outlines the future work.

II. Related Work

Qadir et al. [11] introduced a hybrid deepfake detection model, ResNet-Swish-BiLSTM, which combines convolutional and BiLSTM-based residual networks for training and classification. The model processes sequential video frames to identify artifacts in deepfake images, achieving high accuracy rates—96.23% on the FF++ dataset and 78.33% on combined

FF++ and DFDC datasets. Extensive evaluations demonstrate the method's robustness, generalizability, and superior performance across various datasets, including FF++, DFDC, and Celeb-DF. It mainly focuses on the detection of various deepfake variants: FS, NT, and F2F, with very good performance regarding the discrimination of tampered versus pristine digital footage in terms of recall and AU, reaching up to 0.9876. Future applications of the proposed method in digital forensics are mentioned; future enhancements might be needed with regard to capturing temporal patterns for better adaptability and inference.

In [19], the authors suggested the job of identifying tampered videos on the basis of a hybrid deep learning method by combining CNN and EfficientNet B6. Their proposed framework identifies forged areas by analyzing video frames and was trained on the FaceForensics++ dataset with 90% classification accuracy. The validation metrics considered were precision, recall, and F1-scores of 98%, 81%, and 89% for fake images, and 84%, 98%, and 91% for real images. Apart from that, the trained model was deployed on Flask for real-life testing and validation with the public.

According to [20], it is possible to detect deepfakes effectively using advanced CNN models such as EfficientNet-B4 and XceptionNet. The authors conducted preprocessing through frame extraction and face isolation on the FF++ and Celeb-DF (v2) datasets and trained and tested using the log loss and AUC metrics. EfficientNet-B4 provided an accuracy of 92.99%, followed by XceptionNet with 90.15%, demonstrating strong performance in the classification of real and fake videos. The study emphasized the requirement for continuous updates of detection algorithms to keep up with evolving deepfake techniques.

In [21], a new hybrid transformer network that learns using an early feature fusion method for deepfake video detection was introduced. Their method combines XceptionNet and EfficientNet-B4 as the feature extractors and end-to-end trains them with a transformer on FaceForensics++ and DFDC benchmarks. With a relatively simple architecture, the model achieves performance comparable to existing state-of-the-art techniques. It also employed a new face cut-out and random cut-out augmentation to improve detection performance and avoid overfitting. Furthermore, the research demonstrated good learning capability with limited data. The authors plan to continue their work by training on Celeb-DF and ForgeryNet datasets, testing generalization on unseen samples, and feature use analysis for manipulation detection, such as face swapping and face reenactment.

The study in [5] proposed paper uses a deepfake detection model with the convolutional neural network-EfficientNet, which was trained using the Celeb-DF dataset. This model provided high classification accuracy at 95%, while its recall and F1 scores are 0.9161 and 0.9562 for images with 224x224 pixels, respectively. This closely follows state-of-the-art methods to further prove how solid the approach is. The study therefore highlights the rapid evolution of deep-fake generation techniques, which require adaptable and scalable detection models. Testing on diverse datasets, exploring advanced EfficientNet variants, and increasing training iterations can be done to improve generalization and further improve classification performance.

According to [22], a comprehensive review of deep architectures for deepfake detection shows the transition from CNNs to Transformers. Eight models were evaluated on second- and third-generation benchmarks such as FF++ 2020, Google DFD, Celeb-DF, Deeper Forensics, and DFDC, with accuracies of 88.74% to 99.73% and AUC scores ranging from 97.61% to 100%. The results indicated that CNNs performed effectively in same-train-test settings, while Transformers generalized better across datasets. Further research on the relationships between datasets established the novelty of FF++, Google DFD, and Celeb-DF, as well as emphasized the importance of Deeper Forensics and DFDC in advancing detection methods.

In [23], the authors employed a sequential convolutional neural network and max pooling with the Adam optimizer. It was trained and tested on a combination of datasets, including Celeb-DF and FaceForensics++. The model achieved 93.3% accuracy and 19.5% loss rate, indicating the strengths of CNNs in identifying manipulated content.

The study in [24] proposes a neural network-based approach for the detection of deep-fake videos using advanced deep learning for the classification of genuine and manipulated content. The approach includes preprocessing, feature extraction based on facial landmarks, temporal patterns, and pixel-level inconsistencies, and classification using state-of-the-art neural network architectures such as CNNs and RNNs.

It proposed a diverse dataset of real and synthetic videos, annotated with ground truth labels to ensure robustness and generalization across varied deepfake generation techniques. Extensive experiments are conducted to verify the effectiveness of the framework with high detection accuracy by optimizing the trade-off between computational efficiency compared to the classic GAN-based methods.

This approach further improves scalability and practicality for real-world applications by simplifying the authentication process and reducing computational overhead. This research makes a significant contribution to combating misinformation and securing visual media integrity by providing a computationally efficient and accessible approach to deepfake detection.

In [25], the authors proposed a new Deepfake video detection model, Convolutional Vision Transformer (CViT2), which combines CNNs to extract local features and Vision Transformers to model both global and local relationships of features using attention mechanisms. The model was trained on five datasets: DFDC, FF++, Celeb-DF v2, DeepfakeTIMIT, and TrustedMedia, and tested on 2,669 videos, reaching high accuracy on test sets, including 95% on DFDC, 94.8% on FF++, 98.3% on Celeb-DF v2, and 76.7% on TIMIT.

CViT2 is very efficient in analyzing both pixel-level and non-local features, which makes it more robust for Deepfake video detection under varied scenarios. In the future, more datasets should be included for training to improve generalization and robustness. The proposed model offers a realistic way to counter misinformation and fraud, having

major implications for digital forensics and societal trust in media authenticity.

The work in [2] proposed the challenge of detecting deepfake videos by proposing a model that identifies inconsistencies in facial features, compression artifacts, and manipulation-induced discrepancies. The model, which uses transfer learning on the VGG-16 architecture, trains on the Celeb-DF dataset, focusing on manipulations of facial features for forgery detection. Much emphasis is placed on transfer learning to reduce resource requirements and training time while ensuring robust performance.

The model can extract features that are relevant for deepfake detection, although it has certain limitations with low-quality images and videos. Further improvements, as suggested in the paper, could be made using better datasets, ensemble learning methods, temporal, and audio discrepancies. All these methods will improve accuracy and generalization by combining the results from multiple frames and learning models, hence providing a holistic approach toward deepfake detection.

The study in [26] introduces a method that fine-tunes a transformer module to detect fake images by exploring new feature spaces through attention-based networks, specifically Res-Next CNNs. This architecture emphasizes selectively focusing on critical video features. Frame-level features extracted via Res-Next CNNs are used to train an LSTM-based RNN for classifying videos as real or manipulated. The approach is then validated on various datasets, such as FaceForensics++, Deepfake Detection Challenge, Celeb-DF, and a custom dataset, showing its efficiency in real-time scenarios.

The proposed system has practical implications, such as preventing the spread of misinformation by restricting deep-fake content on social media, news platforms, and law enforcement applications, thus safeguarding the authenticity of online content.

The study in [1] addresses the challenges posed by deep-fake technology by introducing a detection framework based on integrated Vision Transformer architectures, Deep-ViT and Cross-ViT. These models analyze pre-extracted facial features from the FF++ dataset, effectively identifying real and fake faces through subclass-specific detection for manipulation methods and overall classification across all types. The model excels in detecting FaceSwap manipulations, achieving an accuracy of 98%.

Deep-ViT and Cross-ViT leverage the unique capability of Vision Transformers to model both local image features and global pixel relationships, unlike traditional CNN-based deepfake detection approaches. The multi-stream design captures varying scales of alterations, enhancing robustness. The framework's performance was evaluated under intra-dataset and inter-dataset settings, achieving AUC scores of 92.4% on FF++ and 83.1% on Celeb-DF (V2). In subclass detection for FF++ manipulations, the model achieved classification accuracies of 98.6% for Deepfake, 98% for FaceSwap, 97% for Face2Face, and 90.3% for Neural Texture.

This research recognizes advanced architecture's contribution to fighting against the proliferation of deepfakes while highlighting their capabilities to provide safety to the authenticity of media content in view of growingly sophisticated manipulation techniques.

In [27], EfficientNet B7-a state-of-the-art CNN-for detecting deepfake videos. Deepfake techniques that use advanced machine learning to manipulate visual content have created a significant threat to media authenticity. The research will study how EfficientNet B7 can find minute visual cues that hint at manipulation and assess its accuracy, computational efficiency, and robustness on different deepfake datasets. The model achieved an accuracy of 85%, which aligns with the 84.4% accuracy reported in the original EfficientNet paper. EfficientNet B7's efficient architecture and lower computational demands make it suitable for real-world deployment, especially in resource-constrained environments. The study also suggests further research into ensemble models and advanced techniques like knowledge distillation to enhance performance. Eventually, EfficientNet B7 has a very bright future in detecting deepfakes and developing further robust and scalable solutions against misinformation.

The work presented in [3] proposed a neural network-based deepfake video detection by fusing Convolutional Neural Networks and Recurrent Neural Networks. The CNN extracts frame-level features, which are further fed into an RNN, specifically Long Short-Term Memory, to find temporal irregularities and classify whether tampering has occurred in the video. The proposed approach considers two different methodologies of deepfake creation: GANs and autoencoders. Competitive performance is given using ResNext CNN on frame-level detection and RNN on video classification for the detection of deepfake content. The model does even better on real-time data, gaining more accuracy and confidence in classifying a video as real or fake.

The study in [28] deals with the increasing problem of video forgery or deepfake media, which is posing quite a serious threat to media integrity because it is easy to fabricate and disseminate manipulated content. Applications for video forgery detection span across multimedia forensics, digital investigations, and video authenticity verification. The proposed technique relies on a CNN using a ResNet architecture to discover deepfake videos by deep feature extraction from frames. These features are analyzed using sequence descriptors to capture temporal information, which is then processed through fully connected layers to classify videos as real or fake. It is an effective preliminary solution for deepfake detection, based on the DeepFake video dataset, and performance could be improved by updating the data in the real world.

In [29], the authors raise a very critical issue in video forgery and deepfake media, the integrity of which is seriously compromised by the rampant creation and dissemination of manipulated content. The proposed technique utilizes a CNN based on ResNet for deep feature extraction from video frames, together with sequence descriptors that capture temporal inconsistencies. Then, fully connected layers classify videos as either authentic or fake. This approach gives promising

accuracy with the DeepFake video dataset and is the initial framework of deepfake detection. Further enhancement of robustness can be done by training the model with more varied and unseen manipulation techniques. It opens a way for a resilient solution against fake media.

In [30], the authors focus on the advanced technique of deep-fake image detection and propose an improved approach using computer vision and deep learning. It uses the FaceForensics++ dataset and has attained a very high AUC score. The faces are extracted from videos using RetinaFace, and the extraction is accelerated by multiprocessing. Enhancement and augmentation of the dataset are performed, and three models, namely SEResNeXt-50, EfficientNet B0, and EfficientNet B3, are trained and evaluated. Of these, occlusion and blending techniques work the best in combination. This deepfake detection system gives a robust pipeline for verifying video authenticity and is very useful in combating fake news, along with mitigating its potential harms during critical times. Table I provides a summary of some previous works.

TABLE I. RESULTS OF SOME PREVIOUS WORKS

Model	Dataset	Acc
EfficientNEt [11]	DFDC	92.4
EfficientNet-B4[20]	FF++	92.99
XceptionNet[20]	FF++	90.15
No Augs[21]	FS	92.85
No Augs[21]	Deepfakes	95.71
No Augs[21]	F2F	93.57
No Augs[21]	NT	85.00
No Augs[21]	Pristine	96.42
Face cut-out [21]	FS	96.42
ResNet-18[11]	FF++, NT	86.6
Face cut-out [21]	NT	90.71
Face cut-out [21]	Pristine	95.00
ResNet-18[11]	FF++, F2F	92.5
Random cut-out[21]	NT	92.14
Rossler et al [31]	Deepfakes	92.48
Rossler et al [31]	Face2Face	91.33
Rossler et al [31]	FaceSwap	92.63
Rossler et al [31]	NeuralTextures	85.98
CViT2 [25]	DFDC	95.00
CViT2 [25]	FF++ Deepfake	91.26
CViT2 [25]	FF++ NeuralTextures	86.00

III. METHODOLOGY

This section details the methodological setup for developing a robust deepfake detection system, regarding data collection, preprocessing, model architecture, and experimental setup. The proposed methodology is tailored to ensure high-quality input data generation, enabling thorough evaluation and ensuring reliable effectiveness. Fig. 1 shows the pipeline, which outlines the overall key steps.

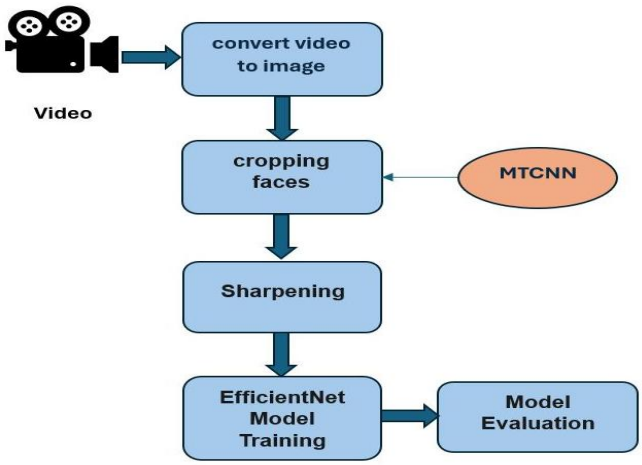


Fig. 1. The architecture diagram of the proposed methodology.

A. Data Collection and Preprocessing

The quality of the dataset and, in many ways, its diversity have large impacts on deep learning model performance. To this end, a large-scale diverse dataset is prepared, including videos from DeepFake-TIMIT, FaceForensics++, Google Deep Fake Detection (DFD), Celeb-DF, and the Facebook Deepfake Detection Challenge (DFDC). Combined, these datasets contained 134,446 videos of about 1,140 unique identities using over 20 different synthesis methods. This large collection enabled broad coverage of deepfake creation methods, significantly enhancing the model's generalization capability.

B. Dataset Preparation

The dataset was carefully analyzed and prepared to ensure its quality and equilibrium. The initial distribution of the frames was heavily unbalanced, with real faces totaling 11 frames and fake faces totaling 50. After preprocessing, it was balanced by fake frames to equal the number of real frames. All images were resized to a standardized format. Steps taken to do this included the removal of all frames where a face could not be detected and low-confidence detections to ensure the quality of the dataset. The preprocessing steps produced a polished, fair, and representative dataset, thus laying a good foundation for model development and evaluation.

This experimental setup fulfilled state-of-the-art best practices and, therefore, ensured reproducibility and utility for future research studies. This approach allows for reliable performance analysis and demonstrates feasibility even in resource-constrained computational environments. Table II provides a summary of used datasets.

TABLE II. SUMMARY OF USED DATASET

Class	Number of images	Dataset	Source	Image size
fake	1100	FaceForensics++	Kaggle	224 ×224
real	1078	DFDC/DFD/ Celeb-DF		

Preprocessing was one of the most important steps in preparing data for model training. Frames were systematically extracted from videos, and their resolution was adjusted based on specified parameters so that the inputs could be standardized. Videos with width less than 300 pixels were upscaled by a factor of 2, those between 300 and 1000 pixels were retained at their original size, those between 1000 and 1900 pixels were downscaled by 0.5×, and videos exceeding 1900 pixels were downscaled by 0.33×. Following frame extraction, face detection and cropping were done using a pre-trained MTCNN algorithm.

Multi-task Cascaded Convolutional Neural Network (MTCNN) is a face detection and alignment deep learning algorithm, which is accurate and highly robust against varying conditions like lighting and pose. It contains three cascaded networks: P-Net (Proposal Network) for generating preliminary face bounding boxes, R-Net (Refine Network) for removing false alarms and refining the boxes, and O-Net (Output Network) for refining the final bounding boxes and facial landmark localization. MTCNN's multi-task learning enhances performance by synchronized face detection and landmark detection, making it efficient and reliable. It is used for face detection in images/videos, facial recognition, real-time face tracking, and facial analysis. MTCNN's cascaded structure ensures high accuracy and is now a cornerstone in face detection research, as illustrated in Fig. 2 and Fig. 3, which shows an example of an image cropped using MTCNN.

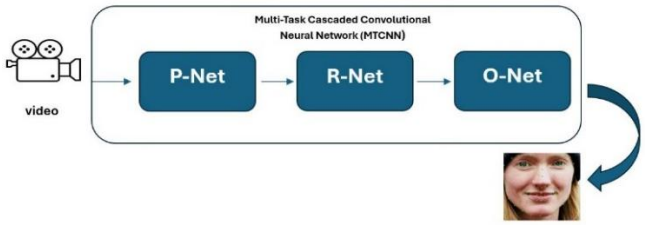


Fig. 2. The architecture diagram of MTCNN.

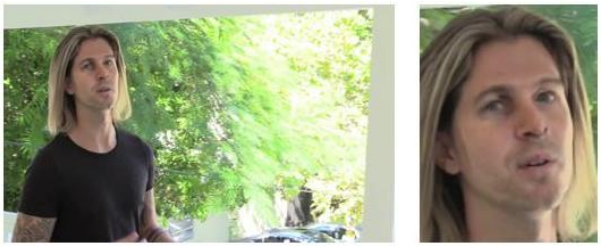


Fig. 3. Example of images cropped with MTCNN model.

In addition, a 30% margin around the bounding boxes was added to enhance the quality of the detected faces by including contextual facial features. A 95% confidence threshold was set to ensure high-accuracy detection; otherwise, if no detectable faces were found in a frame, the frame was thrown out of the dataset. If more than one face was detected in a frame, each face was saved separately. An example of this is detected faces, which consisted of bounding boxes like [110, 58, 71, 92] with a confidence score of 0.999996 and [392 ,60 ,48 ,63] with a confidence score of 0.999997.

To address class imbalance, a common occurrence in deepfake datasets, the number of counterfeit samples was reduced to match the number of authentic samples to prevent any possible bias of the model towards the overrepresented class. Subsequently, the preprocessed dataset was divided into training, validation, and test sets in an 80:10:10 ratio, hence setting up a structured approach to model building, optimization, and evaluation. Next, image improvement was done to improve the quality of the data. Precisely, face image sharpening was carried out using an unsharp masking technique.

Image Sharpening Algorithm enhances image definition and clarity by increasing contrast at edges. It does so by highlighting intensity differences between neighboring pixels, thus making edges more distinct, as shown in Fig. 4. Common methods include the Laplacian filter and Unsharp Masking, which entail subtracting a blurred image from the original. Sharpening is widely used in photography, medical imaging, and computer vision to improve visual quality and perception of detail, so that significant features are more visible. It involves blurring the cropped image with a Gaussian blur and then combining the blurred and original images with weighted summation. The enhanced images, which were converted to OpenCV's BGR format for compatibility, were stored for further processing steps.



Fig. 4. Images after sharpening using unsharp masking.

C. Model Architecture

The deepfake detection model had an EfficientNet backbone, which balanced high performance and computational efficiency. Key modifications included:

The proposed deepfake detection system was based on an EfficientNet backbone, chosen due to its best trade-off between computational efficiency and performance. Its input layer was adjusted to fit 128×128×3 images, which significantly reduced computational costs while retaining crucial visual information. The final convolutional layer output of the EfficientNet architecture was routed through a global max pooling layer to summarize spatial information into a single feature vector. Two fully connected layers with ReLU activation functions were appended to the model to acquire richer feature representations. The output layer used the Sigmoid activation function, allowing the model to output a probability signifying whether an input is deepfake (0) or pristine (1).

Regularization techniques were incorporated to increase model robustness and avoid overfitting. Dropout layers were used, and early stopping was implemented to stop training when validation performance no longer improved. This configuration provided a strong foundation for the accurate and efficient detection of deepfakes. Fig. 5 shows the architecture diagram of the proposed model.

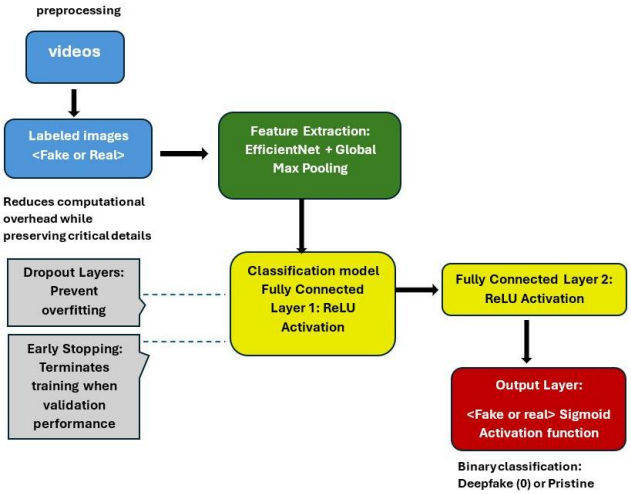


Fig. 5. The architecture diagram of the proposed model.

IV. Experimental Setup

The experimental setup used to train and evaluate the deep-fake detection model is described in detail. The training procedure used the processed dataset with the model, optimized by the Adam optimizer with a learning rate of 0.001, batch size 16, and binary cross-entropy as the loss function. These hyperparameters were tuned to balance convergence speed and model accuracy.

The effectiveness of the model was evaluated using accuracy as the main measure. Accuracy calculates the number of cases predicted correctly to the total number of instances evaluated [32], represented by the Formula (1):

$$\text{Accuracy} = \frac{\text{TP+TN}}{TP+FP+TN+FN} \quad (1)$$

While accuracy is an intuitive and common metric, it can be misleading as a measure of model performance for imbalanced datasets. Nonetheless, in this study, accuracy was the first test of the system's classification abilities that had to be used.

The generalization of the model was tested with a 5-fold cross-validation procedure. It evaluates the model on different subsets of the data, ensuring that it exhibits consistent and robust performance across a variety of training and validation splits.

Preprocessing and training were performed using a CPU-based system, although resource-constrained, thus showing the applicability of the proposed methodology in constrained conditions. The software stack included Python version 3.9 for scripting, TensorFlow version 2.17.1 for model execution and training, OpenCV for image processing operations such as face detection and cropping, and NumPy and scikit-learn for data handling and evaluation. TensorFlow's GPU libraries were installed, but all computational tasks were performed on the CPU, and warnings related to GPU dependencies were ignored.

V. Results

Table III presents the comparative performance of EfficientNet models ranging from B0 to B4, evaluated under identical training and validation conditions. These models were fine-tuned on the dataset with a binary classification task using an input image size of 128×128×128 / times 128×128×128, consistent pre-processing, and training parameters. The main metrics that were evaluated included training accuracy, validation accuracy, training loss, and validation loss, averaging over 20 epochs.

TABLE III. Results of EfficientNet

Model	Validation Accuracy (%)	Validation Loss
EfficientNetB0	94.59	0.13
EfficientNetB1	97.45	**0.04**
EfficientNetB2	96.50	0.08
EfficientNetB3	95.54	0.05
EfficientNetB4	**97.77**	0.13

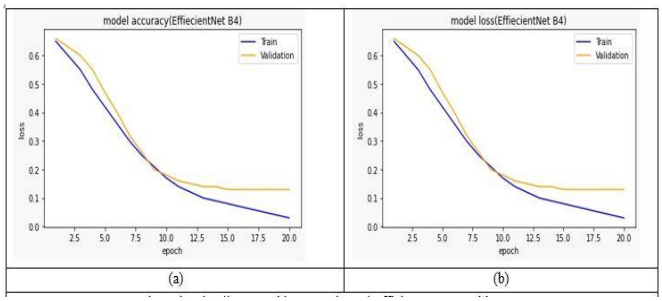


Fig. 6. The plot diagram with 20 epochs and EfficientNet B4 architecture.

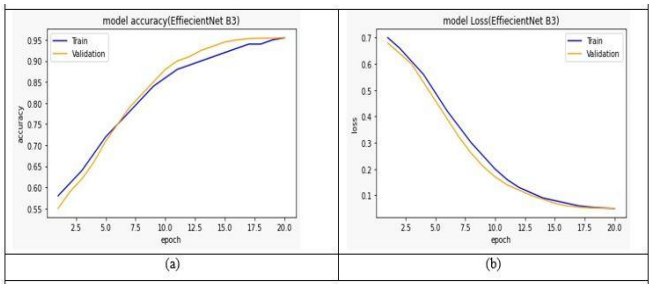


Fig. 7. The plot diagram with 20 epochs and EfficientNet B3 architecture.

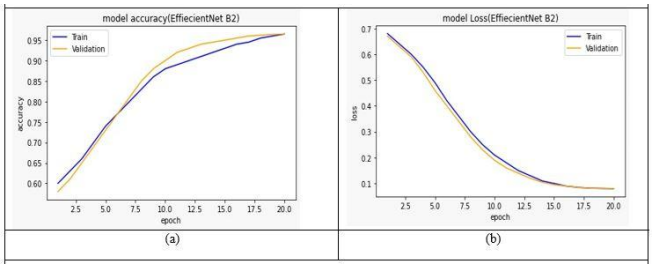


Fig. 8. The plot diagram with 20 epochs and EfficientNet B2 architecture.

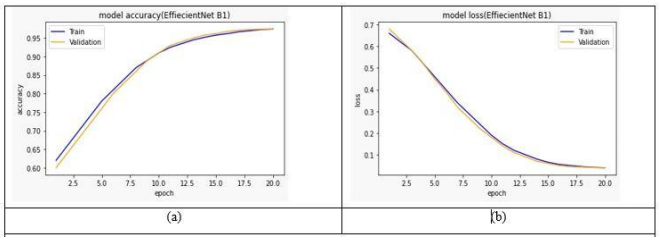


Fig. 9. The plot diagram with 20 epochs and EfficientNet B1 architecture.

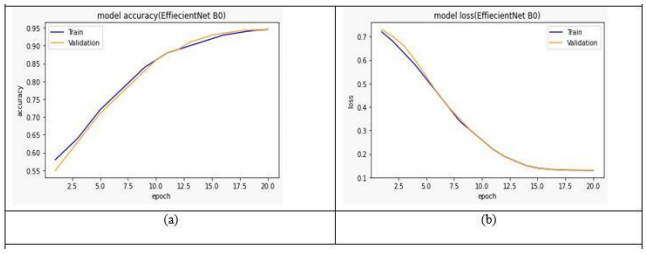


Fig. 10. The plot diagram with 20 epochs and EfficientNet B0 Architecture.

The performance evaluation of various EfficientNet models in detecting deepfakes in videos. Various versions of EfficientNet reveal a variation in validation loss and accuracy, which could be the result of depth and complexity.

EfficientNetB0 reported a validation accuracy of 94.59% and a validation loss of 0.13. The model shown in Fig. 10(a) and Fig. 10(b), although having a reasonably high accuracy, reported a marginally higher validation loss, which reflects a moderate level of misclassification. On increasing the depth of the model, one can observe that the performance significantly increases. EfficientNetB1 performed much better than EfficientNetB0, having an accuracy of 97.45% and a much lower validation loss of 0.04. This demonstrates that increased complexity and parameters result in better feature extraction and generalization.

EfficientNetB2, as presented in Fig. 8(a) and Fig. 8(b), experienced a marginal decrease in accuracy to 96.50% from EfficientNetB1, with the validation loss rising by 0.08. This represents a decrease in predictive efficiency to a marginal degree, perhaps due to overfitting or poor training conditions. EfficientNetB3, as presented in Fig. 7(a) and Fig. 7(b), also maintained a further reduction in accuracy to 95.54%, but its validation loss was low at 0.05. This finding suggests that making the model deeper past a point will not necessarily make it perform better and can instead result in diminishing returns.

Among all the model tests, EfficientNetB4, as illustrated in Fig. 6(a) and Fig. 6(b), achieved the highest validation accuracy of 97.77%, though its validation loss (0.13) was greater than that of EfficientNetB1, as shown in Fig. 9(a) and Fig. 9(b). EfficientNetB4's high performance is due to its better capacity to learn complex patterns from deep-fake videos, leading to stronger feature representation. However, the high validation loss suggests possible overfitting or sensitivity to certain variations of the dataset.

In summary, the results show that increasing EfficientNet models generally enhances classification performance, with EfficientNetB4 being the most accurate. The presence of relatively higher validation loss in certain deeper models shows the trade-off between complexity and generalization. Regularization techniques and data augmentation techniques can be explored further in future studies to minimize the risk of overfitting and further enhance the performance of deepfake video detection.

VI. Discussion

This section provides a critical examination of the above results, with a special focus on evaluating the effectiveness of the proposed deepfake detection technique versus state-of-the-art techniques with EfficientNet architectures. Extra focus is put on identifying how the application of unsharp masking as a preprocessing step aids the model's ability to detect fine visual details characteristic of synthetic video material. By comparison of performance over a number of datasets, this discussion investigates the contribution of both architectural choice and preprocessing method to overall classification accuracy.

To assess the effectiveness of the suggested approach, we compare its performance with comparable works that have applied EfficientNet models to detect deep-fake videos, as shown in Table IV. The results indicate that our suggested models, which employ EfficientNet models and an unsharp masking preprocessing technique, outperform existing methods on several datasets.

The baseline EfficientNet model in earlier studies achieved an accuracy of 92.4% on FF++, DFDC, and Celeb-DF datasets. Similarly, EfficientNet-B4, experimented on FF++ and Celeb-DF, achieved a comparable but slightly higher accuracy of 92.99%. These results, although indicative of good performance, fall short of the proposed models, which incorporate an unsharp masking technique to further enhance feature extraction.

TABLE IV. Comparison between the Proposed Model and the State-of-the-Art

Model	Dataset	Acc
Efficient Net [11]	FF++/ DFDC/ Celeb-DF	92.4
EfficientNet-B4[20]	FF++/ Celeb-DF	92.99
Efficient Net B7[27]	ImageNet	85%
Proposed model EfficientNetB0 + unsharp mask	FF++/DFDC/DFD/ Celeb-DF	94.59
Proposed model EfficientNetB1+ unsharp mask		97.45
Proposed model EfficientNetB2+ unsharp mask		96.50
Proposed model EfficientNetB3+ unsharp mask		95.54
Proposed model EfficientNetB4+ unsharp mask		**97.77**

The proposed EfficientNetB0 + unsharp mask model accuracy was 94.59%, showing the benefit of preprocessing in improving deepfake detection. Most importantly, the proposed EfficientNetB1 + unsharp mask model significantly outperformed the earlier methods, achieving the highest accuracy of 97.45%. This shows that the combination of advanced CNN architecture and image enhancement methods enables the detection of real and synthetic content.

The other proposed models, EfficientNetB2, B3, and B4 with unsharp masking, also performed considerably better than the prior works with 96.50%, 95.54%, and 97.77% accuracy, respectively. The highest performing model, EfficientNetB4 + unsharp mask, achieved a peak accuracy of 97.77%, outperforming the previous uses of EfficientNet architectures on deepfake datasets.

These findings highlight the strength of adding unsharp masking as a preprocessing module, which is likely to enhance edge detection and fine-grained details that are crucial in identifying deepfake artifacts. The improved performance on different datasets suggests that our solution has good generalization capability and can serve as a viable framework for real-world deepfake detection. The study can be extended to analyze the impact of adding other preprocessing modules or hybrid strategies to enhance classification performance.

VII. Ethical Concerns, Data Protection, and Misuse Threat

The rapid development of deepfake detectors, as significant as it is for defending the integrity of digital content, involves a range of ethical, legal, and social challenges. Overcoming these challenges is crucial for ensuring that the design and deployment of such systems are in line with responsibility, transparency, and fairness principles.

A. Ethical Concerns in Model Building

The development of deepfake detectors is accompanied by considerable ethical obligations, particularly maintaining algorithmic fairness. Trainings of such models based on imbalanced datasets have the potential to behave unequally across population subgroups and may exhibit different levels of detection accuracy per gender, ethnicity, or age. These biases can lead to unfair results and must be actively mitigated utilizing heterogeneous and representative datasets in addition to rigorous testing across varied subpopulations [6], [11].

Moreover, the obscurity in deep learning algorithms complicates transparency and interpretability. Because these systems are being utilized increasingly in high-stakes domains such as media verification and forensic analysis, XAI methods need to be incorporated. This enhances user trust, makes one accountable, and gives stakeholders the capacity to understand and audit the system's decision-making better [22].

B. Data Privacy and Consent

Deepfake detectors typically require access to enormous amounts of visual data with identifiable human features. Use of such data raises significant questions about individual privacy and consent. Compliance with data protection law, such as the General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA), requires data collection to be carried out ethically, with explicit consent of individuals involved and appropriate anonymization techniques used, where necessary [6], [7].

Effective data governance frameworks must also be established to secure sensitive information. These encompass safe storage methods, access restrictions, and regular audits to prevent misuse. Ethical review processes must be integrated at all stages in the data life cycle, from acquisition to model deployment, to ensure compliance with privacy protocols and ethics [12].

C. Misuse Risks and Dual-Use Issues

Although the technologies for detecting deepfakes are absolutely crucial for combating digital deception, they possess dual-use potential. Adversaries can use information regarding detection models to develop even more sophisticated forgeries that can bypass existing countermeasures [7], [25]. The constant cat-and-mouse game between detection and forgery can erode confidence in authenticity verification tools among the public.

Distribution of detection tools within sensitive environments—such as journalism, law enforcement, or political communication—also risks misclassification. False negatives or positives can lead to reputational damage, wrongful accusations, or suppression of rightful content. To prevent such risks, the results of detection need to be given with confidence levels and be open to human analysis, particularly in high-risk scenarios [13].

D. Recommendations for Ethical and Responsible Deployment

These social and ethical issues have to be resolved through a collective, multidisciplinary approach from policymakers, lawyers, ethicists, and researchers. Normative legislation should determine permitted uses, designate minimum performance levels, and impose transparency obligations throughout the whole life cycle of deepfake detection tools [6], [27]. Education and awareness campaigns among the public are also necessary to enlighten citizens about the potential and limitations of these technologies.

VIII. Conclusion

This work presents a strong and scalable deepfake detection system that combines EfficientNet models with an unsharp masking preprocessing method to improve the detection of nuanced facial manipulation artifacts. Through training on a large and varied dataset, the system exhibits high performance on various EfficientNet variants, with a best validation accuracy of 97.77% when using EfficientNetB4. The experiments validate the improvement in input image sharpness with unsharp masking as a significant factor in the accurate classification of synthetic versus real facial content. Comparison against previous state-of-the-art approaches validates the effectiveness and stability of the proposed approach against different deepfake generation methods and datasets. The use of deep light-weight EfficientNet models, such as B0 and B1, also offers implementable solutions for deployment under computationally constrained environments without affecting performance. The findings of this study contribute depth to the nascent field of multimedia forensics and digital media authentication. Future studies will tackle combining multimodal features (e.g., audio and temporal features), transparency-driven explainable AI techniques, and real-time implementation strategies to make them practicable and ethical to use in high-risk environments like journalism, law enforcement, and social media.

IX. FUTURE WORK

Future directions will focus on achieving greater robustness and generalization of the deepfake detector by incorporating diverse datasets, including more state-of-the-art deep-fake synthesis techniques. Furthermore, architectural explorations deep into vision transformers and hybrid models capable of encoding spatial and temporal features are expected to maximize accuracy during detection. Incorporation of explainability methods will also be attempted to generate insights into model decisions, guaranteeing transparency and trustworthiness. Additionally, future work will be geared towards optimizing computational processes for real-time detection and deployment of the model in real-world settings, including social media and forensic monitoring.

ACKNOWLEDGMENT

The authors would like to thank all who assisted in preparing this work.

FUNDING

The authors extend their appreciation to the Deanship of Research and Graduate Studies at King Khalid University for funding this work through the Small Research Project under grant number RGP1/38/46.

REFERENCES

[1] S. Hussien and S. Mohamed, “DeepFake Video Detection using Vision Transformer,” International Journal of Intelligent Computing and Information Sciences, vol. 0, no. 0, pp. 0–0, Mar. 2024, doi: 10.21608/ijicis.2024.272675.1324.

[2] A. Karandikar, “Deepfake Video Detection Using Convolutional Neural Network,” *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 2, pp. 1311–1315, Apr. 2020, doi: 10.30534/ijatcse/2020/62922020.

[3] N. Patel, N. Jethwa, C. Mali, and J. Deone, “Deepfake Video Detection using Neural Networks,” ITM Web of Conferences, vol. 44, p. 03024, 2022, doi: 10.1051/itmconf/20224403024.

[4] K. Bjerge, H. M. R. Mann, and T. T. Høye, “Real-time insect tracking and monitoring with computer vision and deep learning,” Remote Sens Ecol Conserv, vol. 8, no. 3, pp. 315–327, Jun. 2022, doi: 10.1002/rse2.245.

[5] G. C. Lacerda and R. C. da S. Vasconcelos, “A Machine Learning Approach for DeepFake Detection,” Sep. 2022, [Online]. Available: http://arxiv.org/abs/2209.13792

[6] T. T. Nguyen et al., “Deep Learning for Deepfakes Creation and Detection: A Survey,” Sep. 2019, doi: 10.1016/j.cviu.2022.103525

[7] I. Goodfellow et al., “Generative adversarial networks,” Commun ACM, vol. 63, no. 11, pp. 139–144, Oct. 2020, doi: 10.1145/3422622

[8] P. Zhu, R. Abdal, Y. Qin, J. Femiani, and P. Wonka, “Improved StyleGAN Embedding: Where are the Good Latents?,” Dec. 2020, [Online]. Available: http://arxiv.org/abs/2012.09036

[9] C. Bravo-Prieto, J. Baglio, M. Cè, A. Francis, D. M. Grabowska, and S. Carrazza, “Style-based quantum generative adversarial networks for Monte Carlo events,” Quantum, vol. 6, p. 777, 2022, doi: 10.22331/Q-2022-08-17-777

[10] S. Suwajanakorn, S. M. Seitz, and I. Kemelmacher-Shlizerman, “Synthesizing obama: Learning lip sync from audio,” in ACM Transactions on Graphics, Association for Computing Machinery, 2017. doi: 10.1145/3072959.3073640

[11] A. Qadir, R. Mahum, M. A. El-Meligy, A. E. Ragab, A. AlSalman, and M. Awais, “An efficient deepfake video detection using robust deep learning,” Heliyon, vol. 10, no. 5, Mar. 2024, doi: 10.1016/j.heliyon.2024.e25757

[12] E. Johansson, “Detecting Deepfakes and Forged Videos Using Deep Learning.”

[13] L. Zhang, T. Dunn, J. Marshall, B. Olveczky, and S. Linderman, “Animal pose estimation from video data with a hierarchical von Mises-Fisher-Gaussian model,” 2021.

[14] K. G. Kim, “Book Review: Deep Learning,” Healthc Inform Res, vol. 22, no. 4, p. 351, 2016, doi: 10.4258/hir.2016.22.4.351.

[15] S. Zdonik, S. Shekhar, L. C. Jain, D. Padua, V. S. Subrahmanian, and N. Lee, “SpringerBriefs in Computer Science Series editors.” [Online]. Available: http://www.springer.com/series/10028

[16] G. R. . Kress and Theo. Van Leeuwen, Reading images : the grammar of visual design. Routledge, 2021.

[17] K. O’Shea and R. Nash, “An Introduction to Convolutional Neural Networks,” Nov. 2015, [Online]. Available: http://arxiv.org/abs/1511.08458

[18] A. Chong, “DeepFake-Detect,” GitHub, 2023. [Online]. Available: https://github.com/aaronchong888/DeepFake-Detect.

[19] S. Toochukwu, “DEEPFAKE DETECTION USING CONVOLUTION NEURAL NETWORK,” 2408. [Online]. Available: www.python.com/flask.html

[20] S. M. Elgayar, O. Abdelhameed, H. M. Ebied, and M. Salah, “Deepfake Detection Using EfficientNet and XceptionNet,” 2024, doi: 10.1109/ICICIS58388.2023

[21] S. A. Khan and D.-T. Dang-Nguyen, “Hybrid Transformer Network for Deepfake Detection,” Aug. 2022, [Online]. Available: http://arxiv.org/abs/2208.05820

[22] V. L. L. Thing, “Deepfake Detection with Deep Learning: Convolutional Neural Networks versus Transformers.”

[23] G. Hamdy1 et al., “Deep Fake Detection Through Convolutional Neural Network,” 2022. [Online]. Available: http://www.ijser.org

[24] R. Gore, A. Kharya, D. Sahu, and S. Kabra, “Deepfake Video Detection using Neural Networks.”

[25] D. Wodajo Deressa, P. Lambert, G. Van Wallendael, S. Atnafu, and H. Mareen, “Improved Deepfake Video Detection Using Convolutional Vision Transformer.” [Online]. Available: https://github.com/erprogs/CViT

[26] S. Jeevidha, S. Saraswathi, K. J. B, R. Scholar, and Bt. Student, “DEEP FAKE VIDEO DETECTION USING RES-NEXT CNN AND LSTM,” 2023. [Online]. Available: www.ijcrt.org

[27] N. Jain et al., “International Journal of INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING Deepfake Detection Using EfficientNetB7: Efficacy, Efficiency, and Adaptability.” [Online]. Available: www.ijisae.org

[28] M. M. Sai, “Deepfake Detection with Deeplearning Using Resnet CNN Algorithm.”

[29] A. Rahman et al., “Short And Low Resolution Deepfake Video Detection Using CNN,” in IEEE Region 10 Humanitarian Technology Conference, R10-HTC, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 459–464. doi: 10.1109/R10-HTC54060.2022.9929719

[30] C. J. Xiong, K. O. M. Goh, and T. Connie, “Deepfakes Detection using Computer Vision and Deep Learning Approaches,” Journal of System and Management Sciences, vol. 12, no. 5, pp. 21–35, 2022, doi: 10.33168/JSMS.2022.0502

[31] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, “FaceForensics++: Learning to Detect Manipulated Facial Images.”

[32] S. Tarek, H. M. Noaman, and M. Kayed, “Enhancing Question Pairs Identification with Ensemble Learning: Integrating Machine Learning and Deep Learning Models.” [Online]. Available: www.ijacsa.thesai.org

Analysis of the Possibilities of Using LLM Chatbots for Solving Course and Exam Tasks

Svetlana Stefanova, Yordan Kalmukov
Department of Computer Systems and Technologies, University of Ruse, Ruse, Bulgaria

***Abstract*—With the widespread introduction of new technologies and, in particular, AI in various areas of life, students are increasingly using large language models (LLMs) such as ChatGPT and other similar tools to help them with their academic tasks. By using them, they can improve their productivity, improve their understanding of complex topics, and support their academic work. LLMs are used both in research, information gathering and preparation for exams and tests, as well as for generating ideas, creating code, and more. This study explores the possibility of using ChatGPT, Claude and DeepSeek for solving course and exam tasks. The results of the analysis could serve as a warning signal and motivation for future transformation of student testing and assessment methods. The ability to use AI systems to search, analyze, and summarize large volumes of information should shift the focus of assessment from classical fact-finding and practical performance of elementary tasks to creativity, combinability, and skills for adapting and applying the already gained knowledge.**

Keywords*—Large language models (LLM); artificial intelligence (AI); ChatGPT; Claude AI; DeepSeek; AI in education; AI for solving exams

I. Introduction

The use of large language models (LLMs) and chatbot platforms in educational contexts, especially for solving course and exam tasks, is growing and offers various possibilities, but also raises important ethical and pedagogical questions.

One of the directions of use is for teaching assistance, as learners can ask questions and receive information and explanations. In some of the tasks, the solution can be given step by step, rather than just providing the final results. In this way, students can be supported in the process itself and encouraged to think critically by asking them questions step by step instead of providing them with final answers. Unfortunately, some students rely too much on LLMs and do not even check the information they generate, which prevents them from mastering key skills in solving problems. An additional potential risk for them is the sometimes inaccurate or downright false information generated.

Another direction of use is when working on projects. In this case, logical frameworks would be helpful before starting the writing process itself, detecting and correcting stylistic and grammatical inaccuracies and errors, as well as generating quotes. Greater attention should be paid when such tools are used to solve practical tests. If this opportunity is used for exam simulation and self-assessment, then the usefulness is once again a fact. Cases of concern, however, are when LLMs are used to directly solve exams or online tests, which violates academic ethics. The use of real-time artificial intelligence tools during exams is a violation of academic education policies.

Types of AI-based platforms

The platforms and tools used by students can be divided into general and those that are focused on education.

Common platforms include:

- ChatGPT [1] – used for information gathering and problem solving;
- Claude [2] – suitable for longer documents and reasoning tasks;
- Gemini [3];
- Copilot [4] – already built into Word, Excel and other tools of the Office suite;
- DeepSeek by the Chinese hedge fund High-Flyer [5].

The tools focused on education are Khanmigo by Khan Academy [6]; Google Socratic [7] as a mobile application for explaining test problems; Quillionz [8] for generating test questions, and Elicit [9] for searching and finding academic articles and other publications.

In order to prevent risk, it is advisable to set clear rules for when the use of artificial intelligence tools is allowed, giving priority to originality. It is good for students to understand the strengths and weaknesses of the LLM so that they can be aware of the role of analytical and critical thinking.

This study performs a series of experimental analyses aiming to test whether the LLM chatbots like ChatGPT, Claude and DeekSeek could solve exams in real time without any further interaction, rather than just providing exam materials as an image taken from a smartphone or smart watch. Results revealing the short answer “yes” should serve as a warning signal and motivation for future transformation of student testing and assessment methods.

The study is structured as follows: Section II reviews related work done by other researchers. Section III presents the experimental setup and the examination materials used to test the LLM chatbots. Section IV discusses the experimental results, their alignment with the related work and their impact on the methods of examining and evaluating students. Finally, Section V ends the study with a conclusion, outlining the need for future transformation in education in the age of AI.

II. Related Work

Puthumanaillam et al. [10] tested if LLMs (ChatGPT) can successfully complete an entire bachelor course in Aerospace Control Systems. Not just a single final exam, but an entire course with 115 course deliverables, ranging from multiple-choice questions to complex Python programming tasks and long-form analytical writing. ChatGPT successfully completed the course and earned a B grade (82.24%). Its strongest results are in structured assignments and greatest limitations in open-ended projects [10].

Rytilahti and Kaila [11] investigated the capabilities of LLM ChatGPT (GPT3.5 and GPT-4) tools to solve coding exercises in an Introductory programming course. Their results show that the LLM can indeed be quite effective in solving the coding exercises. Depending on the version of the tool, the selected approach (amateur or already experienced student), and the prompt used, ChatGPT was able to achieve between 63.4% and 86.2% of the course's total number of points. If programming exercises were only considered, then ChatGPT answered 100% correctly to 107 (75.9%) and to 139 (98.6%) of the course's 141 programming exercises [11]. This means that even students with no previous experience in programming can successfully complete programming courses by utilizing freely available tools [11].

VarastehNezhad et al. [12] evaluated the performance of popular LLMs (GPT-4o, Claude 3.5 Sonnet, Gemini 1.5 Flash, Llama-3.1-70B, Mistral Large 2, DeepSeek-V2, and Gemma-2-27B) in answering questions in data structure and algorithm design from the Iranian university entrance exams for master's programs in computer science. They analyzed the accuracy and the length of responses to the exams in 2022, 2023 and 2024. The questions were given to the LLMs in both Persian and English so that the authors could compare the LLMs' performance in both languages. Results indicate that GPT-4o achieved the highest average accuracy (75.0%), followed by Claude 3.5 Sonnet (67.2%) and Mistral Large 2 (64.1%) [12]. As expected, evaluated LLMs perform better in English than in Persian, with GPT-4o having the largest performance gap (81.3% in English vs 68.8% in Persian).

Felicia Burlacu [13] analyzes the patterns in the performance of LLMs, i.e. what factors influence the accuracy of the LLMs' responses. She identified that all matters - the format of the question (multiple-choice or short answer), the length of the question and its type (factual or analytical). According to the results, LLMs perform best (71.42%) on multiple-choice questions of medium length (50 to 100 words) and factual type. Question length is important since a short question may not provide enough details to the LLM, while a long question could confuse it. Expectedly, factual questions get a higher average accuracy (64.28%) than analytical questions (42.85%).

Knowing that LLM chatbots could be used for cheating during exams, Simon Kaare Larsen [14] proposes guidelines and strategies for creating LLM-resistant exams, including content moderation, deliberate inaccuracies, real-world scenarios beyond the model's knowledge base, effective distractor options, evaluating soft skills, and incorporating non-textual information.

III. Experimental Setup

Since we teach in subjects related to Computer Science, Web Programming and Artificial Intelligence, we are interested in how well the LLM chatbots could solve the exam materials in these specific subjects.

The most commonly used exam formats for IT courses are "theoretical" with many questions that expect short open answers or multiple-choice answers, and practical exams, where students are required to write programming code and develop working applications. The first one is used to check students' overall knowledge or common sense in a specific area of science, while the practical exam tries to evaluate students' ability to develop real-life working applications on a computer. During practical exams, students are usually allowed to use any third-party educational resources on the Internet, except AI chatbots.

The exam in Distributed Web Applications is a "theoretical" one containing 18 questions that require short open-ended answers. Students have 1 hour to answer all questions, but they usually do it faster. All questions are printed on A4-sized paper, and students are required to answer with 1 to 3 sentences directly on that sheet of paper. Here are some example questions: "Specify two advantages of orchestration over choreography", or "If you need to guarantee a strictly defined order of execution of web services, will you choose synchronous or asynchronous communication between them?"

The exam in Information Retrieval is similar (see Fig. 1). It consists of 17 questions that require short open-ended answers between one word and 2 to 3 sentences. Students receive all 17 questions printed on A4-sized paper and should write their answers on the same sheet as well. Here are some example questions: "What are the main differences between the Latent Semantic Analysis and the Vector Space Model?", "Which similarity measures could be used to calculate similarity between sets of keywords?" or "What is the difference between flat clustering and hierarchical clustering?"

In contrast to the previous two, the exam in Web Programming is practical and is conducted on a computer (see Fig. 2). Every student receives an individual assignment to develop a working web application from scratch. Examples, templates and partly implemented code from the exercises are allowed to be used as reference. The assignment consists of four sub-assignments that upgrade one another and increase the final grade from "Sufficient D" to "Excellent A". An example assignment is shown in Fig. 2.

For the purpose of our experiments, the blank exam papers or assignments, as received by students, are photographed with an ordinary smartphone and sent to the three chatbots – ChatGPT, Claude and DeepSeek. The answers provided by the AIs are then checked for correctness and evaluated by the authors of this study. We use the same grading system that we apply to our students as well. It is based on the percentage of correct answers as follows:

< 40%, Fail F	70 – 84%, Very good B
40 – 54%, Sufficient D	85%+, Excellent A
55 – 69%, Good C	

Изпит по **„Търсене и извличане на информация"**, ОКС Магистър, 15.04.2025 г.

1. Кои са основните разлики между векторния модел за анализ на текст и латентния семантичен анализ?

2. Чрез кои мерки за сходство може да се изчисли семантична близост (степен на подобие) между два възела в таксономия?

3. Чрез кои показатели (метрики) се оценява точността/адекватността на търсенето?

4. Как може да се прецени дали подредбата на върнатите резултати е правилна?

5. По какъв начин (чрез кои мерки за сходство) може да се изчисли коефициент на подобие между два документа/обекта, описани чрез неподредени множества от ключови думи?

6. Защо описанието на документите чрез таксономия от ключови думи е по-подходящо, отколкото описанието чрез неподредено множество?

7. Как се представят документите (във вид на какво) при векторния модел за анализ на текст?

8. Защо е необходимо изчислените коефициенти на подобие да се нормализират спрямо дължината на векторите?

9. Защо при практическата реализация на векторния модел за анализ на текст всъщност се използва обърнат (инвертиран) индекс, а подобията не се изчисляват чрез векторите на документите?

10. Какви са основните разлики между плоските и йерархичните методи за клъстеризация?

11. Кои са двете основни характеристики, чрез които се изчисляват теглата на думите в документите/заявката при векторния модел за анализ на текст?

12. Защо е желателно да се премахнат семантично незначимите думи от текста преди прилагането на векторния модел или който и да е друг модел?

13. Как се изчислява семантичната близост между два документа при векторния модел за анализ на текст?

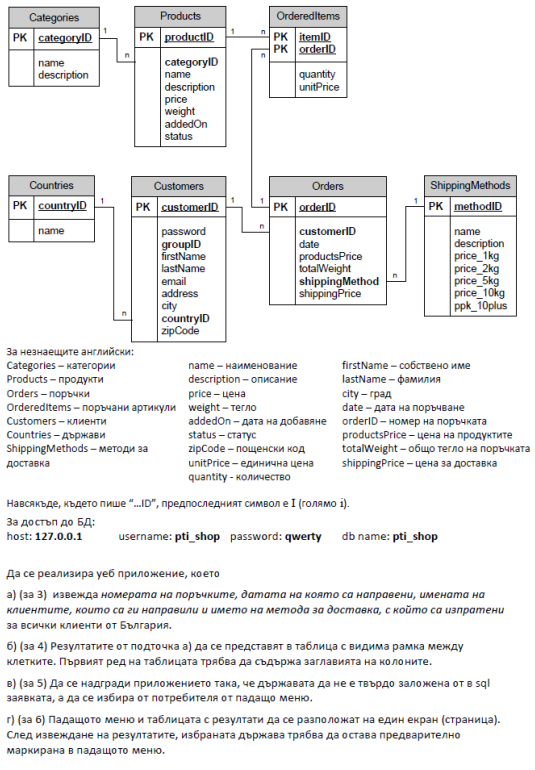
14. Какво представлява операцията стемиране и защо е препоръчително да се извършва преди векторния модел за анализ на текст?

15. Кои фактори влияят върху точността на коефициентите на подобие, изчислени чрез латентния семантичен анализ?

16. Избройте няколко начина, по които може да се изчисли разстоянието между два клъстера?

17. Кой метод позволява откриването и разпознаването на синоними като свързани думи – векторният модел за анализ на текст или латентният семантичен анализ?

Fig. 1. An example assignment for a theoretical exam in information retrieval.



За незнаещите английски:
Categories – категории
Products – продукти
Orders – поръчки
Ordereditems – поръчани артикули
Customers – клиенти
Countries – държави
ShippingMethods – методи за доставка

name – наименование
description – описание
price – цена
weight – тегло
addedOn – дата на добавяне
status – статус
zipCode – пощенски код
unitPrice – единична цена
quantity - количество

firstName – собствено име
lastName – фамилия
city – град
date – дата на поръчване
orderID – номер на поръчката
productsPrice – цена на продуктите
totalWeight – общо тегло на поръчката
shippingPrice – цена за доставка

Навсякъде, където пише "...ID", предпоследният символ е I (голямо i).

За достъп до БД:
host: **127.0.0.1** username: **pti_shop** password: **qwerty** db name: **pti_shop**

Да се реализира уеб приложение, което

а) (за 3) извежда *номерата на поръчките, датата на която са направени, имената на клиентите, които са ги направили и името на метода за доставка, с който са изпратени* за всички клиенти от България.

б) (за 4) Резултатите от подточка а) да се представят в таблица с видима рамка между клетките. Първият ред на таблицата трябва да съдържа заглавията на колоните.

в) (за 5) Да се надгради приложението така, че държавата да не е твърдо заложена от в sql заявката, а да се избира от потребителя от падащо меню.

г) (за 6) Падащото меню и таблицата с резултати да се разположат на един екран (страница). След извеждане на резултатите, избраната държава трябва да остава предварително маркирана в падащото меню.

Fig. 2. An example assignment for a practical exam in web programming.

IV. Results and Discussion

Results from the theoretical exams in Distributed Web Applications and Information Retrieval are shown in Table I and Table II, respectively.

TABLE I Evaluation of the Answers Provided by ChatGPT, Claude and DeepSeek to the Questions from the Exam in Distributed Web Applications

LLM Model	ChatGPT (GPT-3.5)	Claude (Sonnet v3.7)	DeepSeek (V3)
Result, % correct answers	**91 %**	**92 %**	**82 %**
Grade	Excellent A	Excellent A	Excellent A-

Obviously, all three LLM chatbots will get an Excellent A in distributed web applications. All models provide correct answers only, but as seen in Table I, they achieve different percentages. How is that possible? Since there are questions that require a subset of the correct answers, like "Specify two advantages of orchestration over choreography", that makes it possible. There are 4 to 5 advantages of orchestration over choreography, but they have different importance. Some of them are more important than others. DeepSeek, for example, provided two which are correct, but less important, so it does not get full points for this question.

TABLE II Evaluation of the Answers Provided by ChatGPT, Claude and DeepSeek to the Questions from the Exam in Information Retrieval

LLM Model	ChatGPT (GPT-3.5)	Claude (Sonnet v3.7)	DeepSeek (V3)
Result, % correct answers	**95 %**	**99.4 %**	**82 %**
Grade	Excellent A	Excellent A	Excellent A-

Similarly, when solving the exam in Information Retrieval, DeepSeek gives the lowest percentage of correct answers again, while Claude achieves 99.4%. In respect to language clarity, Claude does an excellent job generating beautiful sentences that sound like written by a real human. It should be mentioned here, that all exam materials are in Bulgarian, thus the AI answers are in Bulgarian as well.

The next challenge for the three LLM chatbots is to solve practical exams and write or generate real programming code. It is known that they are good in providing working code fragments or entire basic applications, but it is curious if they can correctly understand our specific assignments and generate working and efficient code. Again, the assignment is photographed with an ordinary smartphone and sent to them for execution. Results are summarized in Table III.

In terms of programming code, all models generate completely working code that satisfies all four sub-assignments, so the chatbots will get an Excellent A grade. ChatGPT and Claude use two additional and unnecessary arrays to cache data from the database, and thus two unnecessary cycles to create these arrays. When told that these two arrays are not necessary,

ChatGPT argues and motivates its decision to cache the data, while Claude agrees they could be omitted.

TABLE III EVALUATION OF THE PROGRAMMING CODE PROVIDED BY CHATGPT, CLAUDE AND DEEPSEEK AS A SOLUTION TO THE PRACTICAL EXAM IN WEB PROGRAMMING

LLM Model	ChatGPT (GPT-3.5)	Claude (Sonnet v3.7)	DeepSeek (V3)
Working code?	**Yes**	**Yes**	**Yes**
Optimal code?	Uses 2 unnecessary arrays and thus 2 unnecessary cycles to create them (but could motivate its decision why)	Uses 2 unnecessary arrays and thus 2 unnecessary cycles to create them	Yes
Grade	Excellent A	Excellent A	Excellent A

The results of our experiments with both theoretical and practical IT exams are similar and confirm those achieved by Puthumanaillam et al. [10] and Rytilahti and Kaila [11] in other subjects.

Since students are illegally trying to use artificial intelligence (AI) during their exams, it is interesting to know how the AI chatbots themselves would evaluate a student who, during a practical exam, has to create an application, has the right to use all Internet resources, without AI, but he or she cannot do anything alone. However, if he or she is allowed to use artificial intelligence, then he or she gets an excellent solution from the AI, but does not understand it.

Claude replies that, in its opinion, the student deserves “Fail F”, because the essence of both education and assessment is to measure students’ knowledge and skills, not their ability to find someone or something to do the work for them.

ChatGPT states that the grade should be “Sufficient D” because we should not only evaluate knowledge, but also the ability to solve problems. If the student can solve problems with the help of AI, then he or she has at least some ability to find a solution.

According to DeepSeek, the fair grade is “Sufficient D” since the problem is "solved", but the lack of understanding is a critical flaw in the educational context. The chatbot adds that, if the student can demonstrate how he or she used the AI, this shows some metacognitive skills and would even justify a “Good C” grade.

Although LLM chatbots seem to easily solve exams, there are other tasks they cannot solve, even if they generate perfectly working programming code for solving them. For example, the assignment problem [15] or other optimization tasks. When asked why they are able to generate a programming code to solve the task but cannot provide the solution directly, LLM chatbots reply that they generate the code as text based on their training, but they do not have an access to servers to run it. That is why they can provide just the code, but not the overall final solution. This means that the programming code they generate for the practical exams has not been tested before giving it to students, and thus the LLM chatbots cannot actually guarantee that it is really working.

V. CONCLUSION

A series of experiments have been conducted, aiming to test whether large language models (LLM)-based chatbots could solve exam tasks in real-time with no or minimum interaction. Results could be summarized as follows:

1) All the three AI chatbots (ChatGPT, Claude and DeepSeek) do an excellent job in solving both theoretical and practical exam tasks in the specified subjects, and would have earned an Excellent A grade.

2) Students are not required to have any special skills in working with AI, nor skills in how to ask questions in order to get an accurate and correct answer. They simply take a picture of the assignment and send it to the chatbot. This could happen illegally during the exam, even without the teacher noticing that.

3) When generating short open-ended answers in text format, Claude does the best job, answering most clearly and purposefully.

4) In general, when generating text in Bulgarian, Claude performs best by providing short, beautiful and human-like sentences.

5) DeepSeek answers the leanest and often tends to omit basic and well-known facts in its answer.

Our experimental results completely align with those of other researchers, showing that LLM chatbots are quite good at solving both theoretical and practical IT exams. Especially when it comes to factual (fact-finding) questions or programming code generation, LLMs will not just pass the exam, but will get an excellent A or B grade.

It seems that it is time to change the educational system again and shift the focus of assessment a little bit from the classical fact-finding and practical performance of elementary tasks to creativity, combinability, and skills for adapting and applying already gained knowledge.

ACKNOWLEDGMENT

This study is supported by the Scientific Research Fund of the “Angel Kanchev” University of Ruse, Bulgaria.

REFERENCES

[1] OPENAI. Introducing ChatGPT, 2022, https://openai.com/index/chatgpt/ (Accessed April 2025).

[2] ANTHROPIC PBC. Meet Claude – the AI for all of us, 2025, https://www.anthropic.com/claude (Accessed April 2025).

[3] GOOGLE DEEPMIND. Gemini – Google’s most intelligent AI models, 2025, https://deepmind.google/technologies/gemini/ (Accessed April 2025).

[4] MICROSOFT. Copilot: Your AI companion, 2025, https://copilot.microsoft.com (Accessed April 2025).

[5] DEEPSEEK, 2025. https://www.deepseek.com/en (Accessed April 2025).

[6] KHAN ACADEMY. Meet Khanmigo: Khan Academy’s AI-powered teaching assistant & tutor, 2025, https://www.khanmigo.ai/ (Accessed April 2025).

[7] GOOGLE. Socratic: Get unstuck. Learn better, 2025, https://socratic.org (Accessed April 2025).

[8] QUILLIONZ. World's First AI-Powered Question Generator, 2025, https://www.quillionz.com/ (Accessed April 2025).

[9] ELICIT. The AI Research Assistant, 2025, https://elicit.com/ (Accessed April 2025).

[10] G. Puthumanaillam, T. Bretl, and M. Ornik. (2025). The Lazy Student's Dream: ChatGPT Passing an Engineering Course on Its Own. arXiv preprint arXiv:2503.05760.

[11] J. Rytilahti, and E. Kaila. (2024). HOW EASY IS IT TO CHEAT?-SOLVING PROGRAMMING EXERCISES AUTOMATICALLY WITH AI. In Proceedings of the 20th International CDIO Conference. Proceedings of the 20th International CDIO Conference.

[12] A. VarastehNezhad, R. Tavasoli, M. Masumi and F. Taghiyareh, "LLM Performance Assessment in Computer Science Graduate Entrance Exams," 2024 11th International Symposium on Telecommunications (IST), Tehran, Iran, Islamic Republic of, 2024, pp. 232-237, doi: 10.1109/IST64061.2024.10843484.

[13] F. Burlacu. (2024). Patterns of success and failure: Analysing Large Language Models in Question Answering in Exam Contexts (Bachelor's thesis, University of Twente).

[14] S. K. Larsen. (2023). Creating Large Language Model Resistant Exams: Guidelines and Strategies. arXiv preprint arXiv:2304.12203.

[15] Y. Kalmukov, “An algorithm for automatic assignment of reviewers to papers”, Scientometrics, 2020, No 124 (3), pp. 1811–1850, https://doi.org/10.1007/s11192-020-03519-0.

Segment-Based Vehicular Congestion Detection Methods Using Vehicle ID and Loss of Expected Time of Arrival

Mustapha Abubakar Ahmed, Azizul Rahman Mohd Shariff*
School of Computer Sciences, Universiti Sains Malaysia, USM, Penang, 11800, Malaysia

***Abstract*—Increasing number of vehicles and rapid urbanization are the significant causes of road traffic congestion. Road traffic congestion is the main issue facing world cities today. Congestion control and mitigation are necessary to mitigate the negative impacts of road traffic congestion, such as delays and increased fuel consumption, among others. There are many congestion detection methods published in the literature; some of these methods, such as the speed threshold, use a single congestion detection metric. Using a single parameter for traffic congestion detection might produce false and inaccurate results. Furthermore, many congestion detection techniques fall short in describing traffic congestion from the user's perspective and vision. To address this, this study develops a segment-based congestion detection method that uses vehicle ID and loss of expected time of arrival. The ID-based method considers both vehicle speed and density, whereas the loss of expected time of arrival focuses on the time loss. These methods are segment-based, where roads are divided into segments using vehicle trajectories. Using a speed threshold of 8.33 m/s, the road is segmented into segments of 8.33 m, 16.66 m, and 24.99 m in length. Vehicle speed and density are monitored using vehicle identification numbers (VINs). Experimental results reveal that the speed threshold and the Microscopic Congestion Detection Protocol recorded false congestion detection. The proposed ID-based congestion detection method is capable of identifying false congestion and accurately detecting real congestion. Moreover, the loss of expected time of arrival shows a promising result in terms of identifying congestion based on motorists' feelings.**

Keywords*—Vehicle ID; traffic congestion; congestion detection; vehicle trajectories; vehicle speed; vehicle density; loss of expected time of arrival

I. Introduction

One of the main issues facing today's urban cities is road traffic congestion. According to [1], it affects social and economic productivity, contributes to fuel consumption and environmental pollution, it adds delays to personal mobility [2]. Congestion leads to wasted time and increased fuel consumption. Loss time leads to reduced productivity.

ITS-V2X technology is regarded as a key enabler for predicting and managing road traffic congestion, as well as enhancing road safety. V2X is an approach to exchanging information between vehicles and infrastructure that utilizes decentralized wireless technologies. Future vehicles will share information about their current position (GPS), speed, and traffic status, among other vehicle details, and utilize this information to optimize their routes.

This information is shared using a message called CAM (Cooperate Awareness Messages). CAM is a standard proposed by the European Telecommunications Standards Institute [3], which enables vehicles to collect their kinematic information and share it with other vehicles and infrastructure. Researchers are now leveraging these free, continuous, and abundant messages to manage, build, predict, and make more informed decisions about road traffic congestion.

It is known that drivers and passengers in moving vehicles measure congestion with visual and temporal perception. The mind of a person in a travelling vehicle, upon seeing a high density of vehicles through the windscreen, would ultimately tell themselves that congestion occurs. Moreover, a person would, in general, continuously check how soon or later he would arrive at the destination, whether on a congested or non-congested road. Regardless of how fast a vehicle is travelling, the person would be more concerned with temporal perception, which involves the perception of time. When motorists are unable to reach their destination on time, the emotional feeling is that there is congestion. Even when vehicle speed and related parameters are given as the primary criteria for evaluating congestion, they are nevertheless inaccurate and misleading. A practical observation on a major road, conducted as part of this study, found that vehicles may travel at speeds exceeding the 30-kilometer-per-hour speed limit and still experience intense congestion.

This research employs vehicle trajectories (GPS coordinates) and vehicle IDs to develop an ID and Loss of Expected Time of Arrival (LETA) method for detecting vehicular traffic congestion. Roads are segmented using the concept of the Cell Transmission Model (CTM) [4]. Using vehicle trajectories and a speed threshold of 8.33 m/s, the road is segmented into three different segment lengths: 8.33 m, 16.66 m, and 24.99 m. The ID-based traffic congestion detection system considers both speed and density. Vehicle speed and density are tracked and monitored on each segment using vehicle ID. LETA is calculated using a vehicle's expected time of arrival (ETA) and actual time of arrival (ATA).

The remainder of the study is organized as follows: Section II discusses the relevant literature. Section III will cover methodology, including road segmentation techniques and congestion detection methods. Section IV compares the performance of ID-based with a speed threshold and analyzes the results. LETA results are also presented in this section. Section V concludes the study and presents future work.

II. Related Work

Trajectory data can provide valuable insights for decision-making when properly examined, revealing patterns in road traffic congestion, urbanization, and traffic. The beginning, middle, and end of the road are all potential locations for traffic congestion. For easy monitoring, the road needs to be divided into segments. Model-based or criteria-based segmentation is the possible ways of segmenting trajectories. The method of segmenting a trajectory into a limited number of segments that must all satisfy a global criterion is known as criteria-based segmentation [5]. Using predetermined criteria, criteria-based segmentation divides a road into segments that satisfy particular spatiotemporal requirements. Each segment must fit a particular model parameter, which is the foundation of model-based segmentation.

Hausdorff distance was employed in [6], a shape-based distance measurement for trajectory clustering, as one of the related efforts documented in the literature about trajectory segmentation. Origin and destination were employed by [7] to categorize trajectories based on business affairs. In [8], the authors broke the trajectory into segments using a similarity score in an attempt to create a trajectory segmentation map that matches large-scale GPS data. A system that divides trajectories into homogeneous segments based on spatiotemporal parameters, such as heading, speed, and position, was created by [9]. In [10], the authors divide the road lane into a segment of cluster cells, each with a 2-meter length. A protocol for evaluating and detecting traffic congestion on a road segment was presented by [11]. They describe a road segment as a road that connects two intersections without specifying its length or size. Although these works divided roads into segments, they did not utilize the concept of CTM.

There are two ways of detecting road traffic congestion: using fixed equipment and floating vehicles [12]. Fixed equipment-based congestion detection involves using cameras, loop detectors, and other devices to detect road traffic congestion. The accuracy of this method is high; however, its coverage is limited to a specific location [13]. Using floating vehicles involves utilizing vehicles equipped with sensors and GPS technology on the road to detect traffic congestion. This method has the advantage of detecting congestion over large areas. It also provides a cost-effective method of analyzing and detecting road traffic congestion [14].

Congestion can be quantified using various metrics, including speed, trip duration, density, and congestion index, among others. GPS-based congestion detection is the most cost-effective and widely used approach. Vehicles can also be easily tracked [15]. Road traffic congestion is measured using a variety of techniques; however, there is no single, universal method to determine the state of traffic. Congestion can be measured using the following criteria: 1) Average speed, 2) Travel time, 3) Delay, 4) Density, 5) Level of service, and 6) congestion indices [16]. According to [17], speed, density, and degree of saturation are good indicators of traffic conditions.

Both stationary equipment and floating vehicles can be used to identify congestion [12]. Detecting traffic congestion on the road by using vehicles fitted with sensors and GPS technology was known as "floating vehicles". The benefit of this approach is that it can identify congestion over vast areas. Additionally, it offers an economical approach for assessing and identifying traffic jams [14]. Many studies have utilized data from floating vehicles to develop a road traffic congestion detection system, which includes methods developed by [18, 19, 13], all of which employ fuzzy logic for V2V congestion detection. However, the system incurs overhead due to the message exchange between vehicles for congestion validation.

Fuzzy logic was also employed by [20] to estimate traffic flow. Vehicle speed and location were captured using RFID and transmitted to the cluster head. The cluster head will calculate the average speed and density and then use fuzzy logic to determine the flow. RFID is expensive to implement for all vehicles on the road, and vehicle clustering is also a time-consuming process.

Vehicle trajectories were segmented based on the time window in [21] to detect moving clusters. Vehicle speed and density were used to detect slow-moving clusters for congestion detection. The system is based on clusters, and the formation of these clusters is time-consuming.

A congestion detection approach using clustering algorithms was developed in [22]. Vehicle trajectories are clustered, and the distance travelled by the clusters is computed to determine the average speed, which measures congestion. This method is based on historical data.

Pollutant emissions and traffic delays can be successfully decreased using vehicle speed guidance. Driving too fast is a significant contributing factor to traffic accidents, and research indicates that speed is a leading cause of deadly traffic accidents in nearly every nation. The effects of setting a speed restriction of 30-kilometer-per-hour on roads have been extensively studied by transportation researchers.

The study by [23] found that enforcing a 30-kilometer-per-hour speed restriction in 40 European towns significantly reduced traffic accidents, injuries, and fatalities. In [24], the authors analyzed the effectiveness of a 30-kilometer-per-hour speed limit on roads, finding that it reduced energy use and saved lives.

According to [25], speed emerges as the predominant metric used in the congestion detection literature. Utilizing a single parameter, such as speed, for traffic congestion detection might not produce accurate results [26], [27]. The speed threshold was proposed by [28]. The vehicle's speed was calculated from the distance it covered, and the average speed was compared with the threshold to determine road congestion. Speed may overlook how traffic congestion develops across space, leading to incorrect identification of traffic congestion.

This study presents an ID-based congestion detection method to address the issue of relying on a single parameter for traffic congestion detection. Using a speed threshold, time headway for congestion detection may be inaccurate, leading to false detections. Travel time and delay may not predict how early or late a motorist will arrive at their destination. This makes motorists question when they will get to their destination. LETA is intended to determine how late or early a motorist will arrive at their destination.

III. METHODOLOGY

A. Road Segmentation

CTM defines cell length as equal to the distance a vehicle travels at free-flow speed [4]. In the speed threshold [28], free flow speed was defined to be 8.33 m/s, which means the vehicle will move 8.33m in one (1) second or above to be considered not in congestion. The use of 1s is driven by the fact that the maximum CAM broadcasting interval is 1 second. In this research, vehicle information is collected every 1 second to determine congestion. A vehicle will drive 8.33m/s in 1s, 16.66m/s in 2s, and 24.99m/s in 3s to be considered congested.

Three different segment lengths are utilized to evaluate the method's effectiveness: 8.33 meters, 16.66 meters, and 24.99 meters. In other words, the road is divided into three segments: 8.33 meters, 16.66 meters, and 24.99 meters. Seongsin-ro 2-gil, 140.06 meters, and part of Gonghang-ro, 101 meters in length, are used for the implementation of the proposed congestion detection methods. Table I shows the lengths of the three segments as well as the number of segments generated.

TABLE I. SEGMENT LENGTHS AND THE NUMBER OF SEGMENTS GENERATED

Segment Length	Number of Segments Created (Seongsin-ro)	Number of Segments Created (Gonghang-ro)
8.33 m	16	12
16.66 m	8	6
24.99 m	5	4

The road is not physically segmented; rather, the ID-based system virtually segments the road based on the vehicle trajectories collected, as illustrated in Fig. 1. Vehicle speed is monitored using the vehicle ID and its trajectory. Vehicles driving on a segment of 8.33-meter length are expected to exit after one (1) second. With a segment length of 16.66 meters, vehicles are supposed to exit the segment after two (2) seconds. Vehicles will take three (3) seconds to exit a segment of 24.99 meters in length. This process allows for the monitoring of your vehicle speed in each segment using three different segment lengths.

B. ID-Based Congestion Detection

The density metric was used in the majority of research investigating traffic congestion by counting the number of vehicles per unit of road length. Our approach uses the vehicle's ID sent in the vehicular messages to identify slow-moving vehicles within a road segment. In terms of this research is the first study to detect road traffic congestion by using vehicle ID to identify slow-moving vehicles.

In essence, a traffic flow refers to a line of moving vehicles on a specific route. Vehicles are grouped into substantially denser clusters, travelling at extremely slow speeds because there is insufficient space for lane changes or passing when there is heavy traffic [21, 29]. Therefore, the motivation is that tracking and identifying the group of slow-moving vehicles can effectively identify traffic congestion.

ID-based is a centralized system. The system uses the vehicle ID and trajectories to determine congestion, as shown in Fig. 1. At each time t (i.e., 1s), the vehicle broadcasts its status messages to nearby vehicles and RSUs. After receiving status messages from vehicles on a segment, the RSU aggregates vehicle IDs. As previously mentioned, roadways are segmented into 8.33-meter, 16.66-meter, and 24.99-meter segments, and congestion is measured separately in each segment. Congestion is checked after one (1) second, that is, after each message receipt cycle, for an 8.33-meter segment length. For a 16.66-meter segment length, the system will check for congestion after two (2) seconds, i.e., two message-receiving cycles. Congestion is monitored after three (3) seconds, i.e., three message-receiving cycles, with a 24.99-meter segment length.

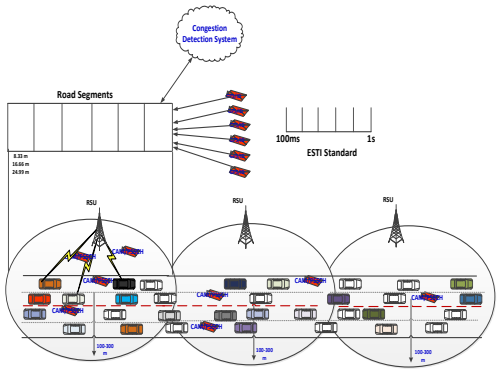


Fig. 1. ID-based congestion detection model.

Let us assume that there are n segments of road Ri. Ri = {Ri1, Ri2, Ri3, …… Rin}. Where i is the road ID, n is the segment ID of road i. Let m be the number of vehicles traversing on Rin such that Rin = {V_1, V_2, V_3, … V_m}, where V represents a vehicle with ID m. Equation 1 shows that all vehicles in Rin can be presumed to be advanced to Ri(n+1) within t seconds (1 second) under light traffic, defined as 30-kilometer-per-hour (8.33 m/s) [see Eq. (1)].

$$R_{i(n+1)(t+1)} = R_{in(t)} \quad (1)$$

Congestion can be detected by detecting slow-moving vehicles [21]. Owing to this notion, an ID-based congestion detection method detects congestion when at least half of the vehicles remain in the segment after the clock tick (1, 2, or 3 seconds, depending on the segment length). It signifies the existence of traffic congestion. This can be evaluated by comparing the vehicle IDs of segment n at time *t* with the IDs of the same segment at time *t* + *i*, where *i* can be 1, 2, or 3, depending on the segment length.

Let Vn (ID) be a vehicle n ID. Let XRin(t) = {V_1(ID), V_2(ID), V_3(ID), … Vn(ID)}, where X_{Rin}(t) is the total vehicle IDs of segment Rin at time t. Congestion occurs if Eq. (2) is valid.

$$X_{Rin(t+1)} \geq \frac{1}{2} X_{Rin(t)} \quad (2)$$

C. LETA

LETA is a segment-based congestion monitoring system. Congestion is tracked on a segment-by-segment basis rather than the road as a whole. It is presumed that RSUs have full coverage of the road. To determine if vehicles may gain or lose time on a specific segment, LETA is computed for each

segment. LETA is capable of monitoring traffic congestion on every segment of the road. LETA is proposed with the intention of estimating the level of traffic a vehicle entering the road would have to experience. A congested road is indicated by loss time, and one that is free-flow is indicated by gain time. The estimated time of arrival (ETA) and actual time of arrival (ATA) must be ascertained in order to compute LETA.

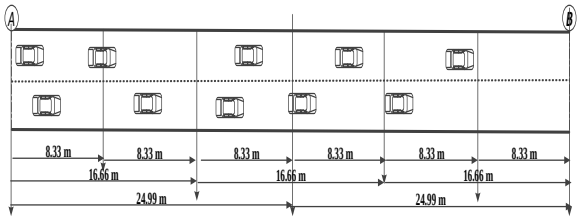


Fig. 2. Shows vehicles moving from A to B across segments.

Let Ri be a road with nth segments, i.e., Ri = {S1, S2, S3, ….., Sn}, where Sn is a road segment with index n of Ri. ETA is the estimated time a vehicle will arrive at destination B from origin A, as shown in Fig. 2. To compute ETA, the distance between two points must be defined. In this work, three different segment lengths are defined: 8.33 m, 16.66 m, and 24.99 m. Using Eq. (3), the ETA can be computed for each segment.

$$ETA_{S_i} = {DSL_{S_i}}/{VS_{S_i}} \quad (3)$$

where, ETASi is the average expected time of arrival for segment Si, DSLSi is the segment length, and VSQi is the segment's average vehicle speed. ATA represents the vehicle's movement on the freeway at the recommended vehicle speed. Using Eq. (4), ATA can be calculated:

$$ATA_{S_i} = {DSL_{S_i}}/{RVS_{S_i}} \quad (4)$$

where, ATASi represents the actual time of arrival for segment Si, DSLSi represents segment length, and RVSSi is the recommended vehicle speed for segment Si. LETA for a segment determines whether there is congestion and how late the vehicle is expected to arrive. LETA is determined by subtracting ETA from ATA and can be computed using Eq. (5).

$$LETA_{S_i} = ATA_{S_i} - ETA_{S_i} \quad (5)$$

where, LETASi is the time loss from the actual time of arrival for segment Sj, ATASi is the actual time of arrival, and ETASi is the expected time of arrival. The total loss of expected time of arrival for road Ri is the sum of the LETA for each of the road's segments. Eq. (6) can be used to calculate the total time loss for road Ri.

$$TLETA_{R_i} = \sum_{i=0}^{n} LETA_{S_i} \quad (6)$$

Where TLETARi is the total time loss for road Ri, LETASi is the loss of expected time of arrival for segment Si, and n is the total number of segments in Ri.

IV. Evaluation of Approaches

A. Dataset

Jeju vehicular traces [30] were used. Seongsin-ro 2-gil and Gonghang-ro road are selected for implementation of the ID-based and LETA methods for vehicular traffic congestion detection. Jeju dataset was chosen because it has CAM data properties, and its transmission timing is consistent at (1) second intervals. Seongsin-ro 2-gil is a one-way road measuring 140.06 meters. At the same time, a part of Gonghang-ro, which is 101 meters in length, is also used. These routes were chosen since they have a substantial number of trajectories, and their GPS coordinates do not deviate from the actual road map. The Jeju dataset contains kinematic information collected from vehicles, with 5000 seconds of heterogeneous vehicle simulation traces totaling 8,495,739 traces. The dataset is generated using the SUMO simulator.

B. Data Preprocessing

Working with GPS trajectories requires preprocessing, as some of the GPS points may deviate from the actual road map. Inaccurate results could arise from anomalies and mistakes if preprocessing were not done [8]. Seongsin-ro 2-gil and Gonghang-ro GPS points (trajectories) were matched on Google Maps. The actual road and the GPS position coincided.

C. ID-Based Results

1) Seongsin-ro: To evaluate the effectiveness of the ID-based congestion detection method, a comparison was made with the speed threshold [28] and the microscopic congestion detection protocol (MCDP) [31]. MCDP used vehicle information to calculate the number of vehicles and the time headway between them. If the time headway is less than two seconds, congestion occurs. At the same time, the ID-based method calculates the vehicle's speed based on the distance it covers and compares it with the speed threshold, i.e., 8.33 m/s.

Since the ID-based method was developed to determine whether congestion exists, the number of detection points was used as a metric; the number of detection points is the total number of points where congestion has been identified. This enables a comparison of two congestion detection systems to determine which one has the highest detection rate.

Fig. 3 illustrates the comparison results for an 8.33-meter segment length. The results reveal that both the ID-based speed threshold and the headway-based approach recorded similar patterns. It can be observed that both ID-based and speed thresholds recorded almost the same congestion detection values for all 16 segments. At the same time, a little difference was observed with the headway-based detection method.

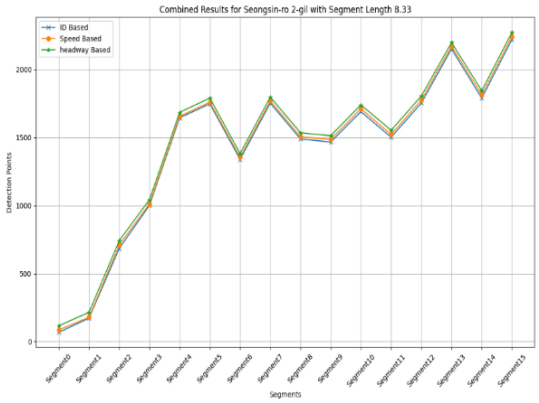


Fig. 3. Shows a comparison between ID-based, speed threshold, and time headway-based results for an 8.33-meter segment length.

Fig. 4 compares the results of ID-based, speed threshold, and headway-based methods using a segment length of 16.66 meters. Both methods recorded a detection point of less than 200 at the first segment, i.e., segment 0, with the ID-based and speed threshold methods being slightly lower than the headway-based method. In both the eight (8) segments, it is clearly shown that ID-based and speed threshold recorded lower detection compared to the headway-based method. It is also notably clear that the ID-based method recorded low detection in comparison to the speed threshold, especially from segment two onward, even though the difference is minimal. This means the base congestion detection method can detect actual (true) congestion.

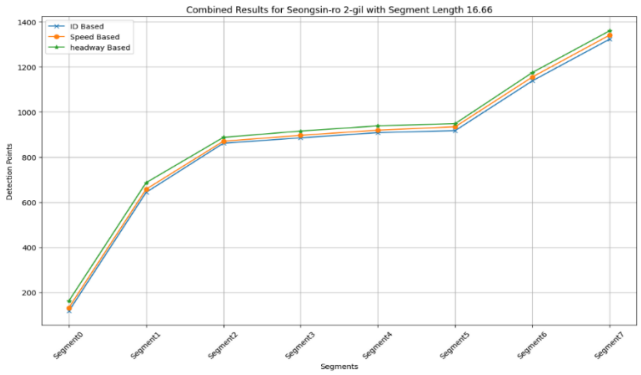


Fig. 4. Shows a comparison of results of headway-based, speed threshold, and ID-based for a 16.66-meter segment length.

Fig. 5 shows a comparison of the ID-based speed threshold and headway-based results for a 24.99-meter segment length. As can be observed, only at the segment0 speed threshold did the detection point go below that of the ID-based method. Even though the difference is minor, in the rest of the segments 1 to 3, it is seen that ID-based recorded lower detection than speed threshold and headway-based.

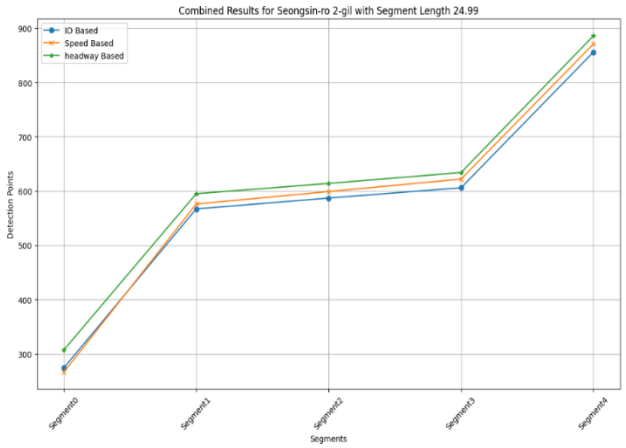


Fig. 5. Shows a comparison of headway-based, speed threshold, and ID-based results for the length of a 24.99-meter segment.

2) Gonghang-ro road: To validate the effectiveness of the base congestion detection method, another road was also considered. Vehicular traces from the Gonghang-ro road are also used to evaluate the ID-based detection method. Fig. 6 compares ID-based results with speed threshold and time headway-based for the 8.33-meter segment length. Congestion detection points are shown for each segment. Segment 0 marks the beginning of the road for all segment lengths.

It can be seen from the results in Fig. 6 that the detection points fluctuate over the segments, with segment 11 recording the lowest values for all the detection methods. The results reveal that with an 8.33-meter segment length, both methods observe the same pattern, with the headway-based method reporting higher detection points than the ID-based and speed threshold. Though the difference between ID-based and speed threshold is not significant, ID-based reported lower detection points.

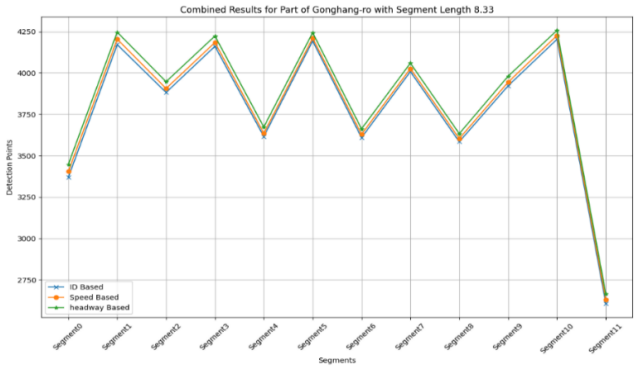


Fig. 6. Congestion detection comparison of the three detection methods using segment lengths of 8.33-meter.

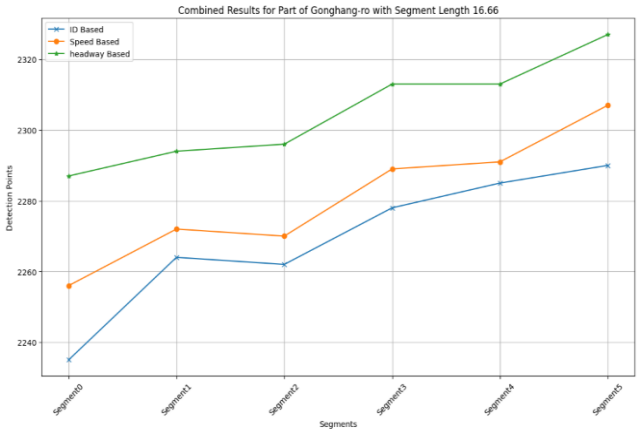


Fig. 7. Congestion detection comparison of the three detection methods using segment lengths of 16.66-meter.

The results in Fig. 7 showed a steady increase in detection points from the first to the last segment for the headway-based method, while ID-based and speed threshold observed a sharp drop in segment 2, then continued to rise till the last segment. With a 16.66-meter length, a clear varying result was observed. Headway has the highest detection points for all six segments, followed by speed threshold, and ID-based has the least.

Similar to results with 16.66-meter segment length, results with 24.99-meter segment length also recorded a drop of detection points on segment 1, as shown in Fig. 8. It was observed that there is a significant difference between the three congestion detection methods in terms of detection points for a 24.99-meter segment length in all four segments.

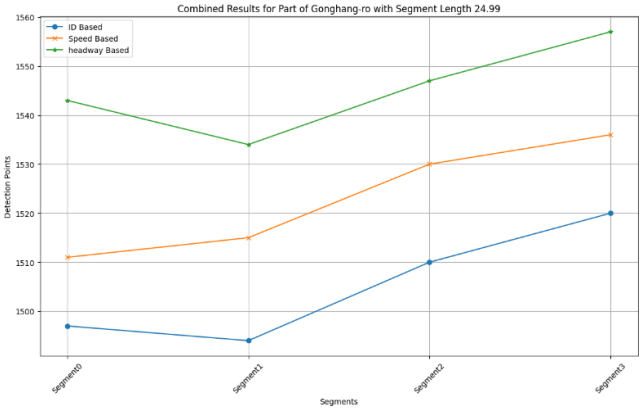


Fig. 8. Congestion detection comparison of the three detection methods using segment lengths of 24.99-meter.

Overall, the results presented show that the ID-based road traffic congestion detection system performed better than the speed threshold (Ahmed, Shariff, and Abubakar, 2024) and headway-based (Ahmad, Chen, and Khan, 2018) in identifying actual congestion in all three different scenarios (8.33m, 16.66m, and 24.99m). ID-based creates a two-layer check by combining density estimation with the speed threshold technique. The ID-based approach employs density estimates to identify congestion that is not bogus congestion after verifying it with a speed threshold.

D. LETA Results

A comparison of the three scenarios—at 8.33meters, 16.66meters, and 24.99 meters—is shown in Fig. 9. The findings indicate that while ETA and ATA are all influenced by speed and the distance between the source and the destination, segments with a length of 24.99 meters have the most significant LETA values in all the segments. The longer the segment distance, the greater the density of the vehicles. Segment 0 has the lowest LETA value in the 24.99 and 16.66 scenarios, as compared to the 8.33-meter scenario, where segment 1 has the lowest LETA value.

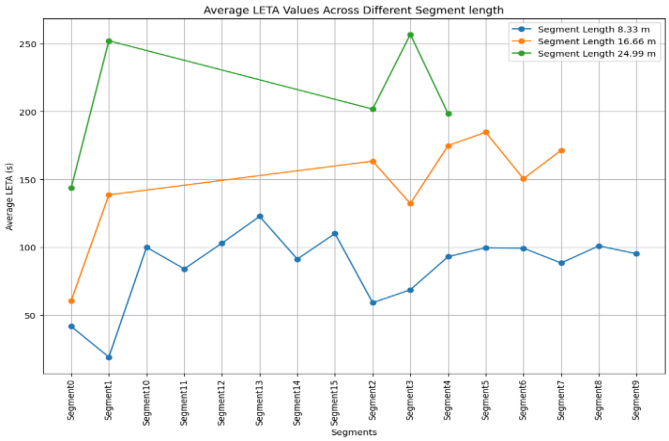


Fig. 9. LETA results over segments for all three scenarios.

A fine-grained LETA study is made feasible by calculating LETA for ten carefully selected vehicles, ensuring that the analysis focuses on specific entities of interest. To produce a combination of the most and least crowded vehicles, ten (10) vehicles were chosen, five with the most traces and five with the fewest. The 10 chosen vehicles are tested in 24.99-meter segments.

Fig. 10 shows the LETA findings for ten (10) chosen automobiles during the simulated period. According to the findings, the LETA of several vehicles, including the JTr450, JBr450, JTx973, JPs973, and JTr832, grows dramatically over time. These are the vehicles with the most traces, and the spikes represent occasions when the vehicles lose a significant amount of time owing to traffic congestion they encountered throughout time. The 5000 to 15000 second LETA spikes observed by several vehicles may represent periods when most vehicles lose time due to traffic or driver behavior. These spikes can indicate specific places or periods of significant traffic congestion.

Throughout the observed time period, vehicles such as JPs967, JTx688, and JPs688 consistently displayed low LETA values, implying that they did not waste much time and most likely moved faster in the absence of significant traffic congestion. All vehicles exhibit low LETA values toward the end of the time period, indicating that they either arrived at their destinations, exited congested regions, or experienced less traffic near the end of their journey.

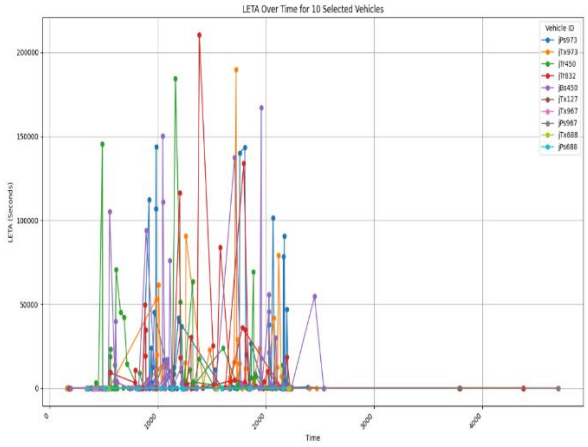


Fig. 10. LETA results of 10 selected vehicles for the entire simulation time.

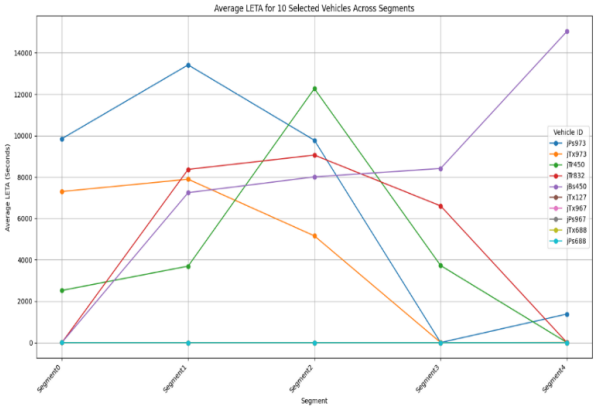


Fig. 11. LETA results of 10 selected vehicles over the segment.

LETA results of the ten (10) selected vehicles are plotted against segments, as illustrated in Fig. 11. The graph depicts the average LETA measurements for 10 distinct vehicles over five (5) road segments of length 24.99. The figure depicts how each vehicle's LETA patterns vary across the segments. Five of the vehicles had a high LETA value, indicating a significant loss of expected arrival time, whilst the remaining five had a continuously low LETA value, indicating smooth movement with little time loss.

Fig. 11 also shows that JPs967, JTx688, JPs688, JTx127, and JTx967 vehicles have a flat trend or almost no decrease in expected time of arrival throughout all segments, with consistently low LETA values, indicating that these vehicles met less traffic on all road segments. Furthermore, Fig. 11 shows that some segments, particularly segments 1 and 2, have higher average LETA values for numerous vehicles, implying that these segments are congested.

Five vehicles with the lowest LETA values and five vehicles with the highest LETA values were analyzed independently to further analyze the results displayed. Fig. 12 depicts the average LETA for five vehicles chosen, as having the lowest LETA along various road segments.

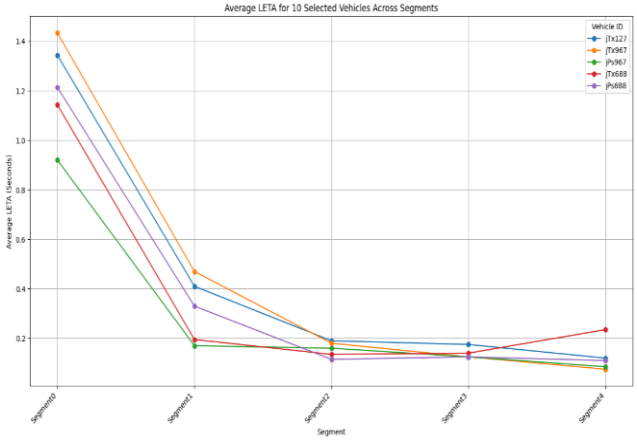


Fig. 12. LETA results of 5 selected vehicles with the least traffic traces.

Fig. 12 shows that each vehicle's LETA decreases significantly from segment 0 to segment 2. This suggests that traffic conditions improved throughout the route since the vehicles' LETA reduced as congestion lessened along the road segments. Only segment 0 of the four vehicles had a LETA value of less than 1 second, while JPs967 had a LETA value greater than 1 second. Between segment 0 and segment 1, the LETA for all vehicles declined by 0.47, and from segment 2 to segment 4, the LETA values for most vehicles remained reasonably steady.

At segment 0, JTx967 has the highest LETA; however, after segment 1, it drops rapidly and stabilizes. Vehicles JTx688 and JPs688 follow similar trajectories, with LETA from Segment1 stabilizing at around 0.2 seconds. JPs967 rises somewhat in Segment 4, but JTx127 and JPs967 continue their downward trajectory.

Overall, the decrease in LETA for each vehicle from segment 0 to segment 1 indicates that the vehicles first encountered traffic, which then subsided after segment 1. In other words, when there was traffic, vehicles moved onto the road, and the bottleneck ultimately disappeared.

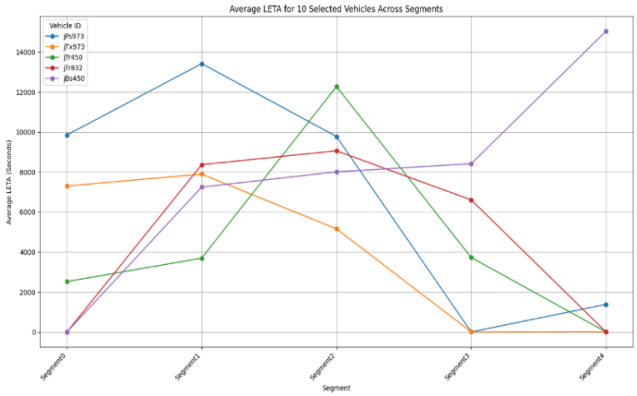


Fig. 13. LETA results of 5 selected vehicles with the most traffic traces.

Fig. 13 displays the average LETA for segment 0 to 4 for the five most congested vehicles (JPs973, JTx973, JTr450, JTr832, and JBs450). The findings shed light on how these vehicles lose LETA when traveling through various parts, whether due to traffic bottlenecks, poor road conditions, or driver conduct.

Fig. 13 indicates that the LETA trend for JBs450 cars is increasing steadily from segment 0 to segment 4, with a particularly noteworthy leap between segment 3 (8413s) and segment 4 (15034s). JTr450 began in segment 0 with a reasonably high LETA value of 2523s, increased to 12271 in segment 2, and then dropped to a low of 7 seconds in segment 4. Vehicle JTr832's LETA grows from 2 seconds in segment 0 to a maximum of 9058 in segment 2, following which it rapidly declines in segment 3 to 4, with segment 4 recording a LETA of 5 seconds.

In segment 0, the JTx973 vehicle had a high LETA value of 7299s. In segment 1, it increased to 7894s, whereas in segment 4, it reduced to 16 seconds. In segment 0, JPs973 has the highest LETA value of 9851s. In segment 1, it climbs to 13416s, whereas segment 3 and segment 4 dip and rise by 3s and 1381s, respectively. Some vehicles experienced increased LETA as they progressed, while others saw LETA drop at the end of their journey. The trend inconsistencies in Fig. 13 suggest that vehicles suffered LETA at different times and segments.

TABLE II. LETA VALUES FOR ALL TEN VEHICLES ACROSS THE SEGMENTS AND THE TOTAL LETA

	Segment0	Segment1	Segment2	Segment3	Segment4	Total LETA
jPs967	0.92	0.17	0.16	0.125	0.085	1.46
jTx688	1.143333	0.195	0.135	0.14	0.235	1.85
jPs688	1.213333	0.33	0.115	0.125	0.11	1.89
jTx127	1.343333	0.41	0.19	0.175	0.12	2.24
jTx967	1.433333	0.47	0.18	0.125	0.075	2.28
jTx973	7299.101	7894.033	5157.802	6.00875	16.84625	20373.79
jTr450	2523.266	3694.31	12271.22	3724.643	7.425	22220.86
jTr832	2.32	8368.568	9058.859	6604.972	5.02	24039.74
jPs973	9851.517	13416.29	9776.06	3.063333	1381.45	34428.38
jBs450	5.62	7243.885	8006.662	8413.615	15034.56	38704.34

Table II displays the LETA values for each segment of the ten vehicles, as well as the total LETA for the entire trip. As previously stated, the table shows that the LETA values for vehicles JTx973, JTr450, JTr832, JPs973, and JBs450 are significantly higher than those for the other vehicles. While the remaining vehicles maintain LETAs below three seconds across all segments, some vehicles have LETAs in the thousands of seconds.

The table also shows that, with a total LETA of 38,704.34 seconds, vehicle JBs450 has the most significant overall time loss across all segments. Surprisingly, the LETA of 15,034.56 seconds is very high in segment 4. Vehicle JPs967, on the other hand, has the least loss of expected time of arrival throughout all segments, with a total LETA of only 1.46 seconds.

E. Discussion

The ID-based results presented showed great improvement compared to MCDP and speed-based congestion detection methods. This is because of the ID-based congestion detection method check vehicle speed and density in order to ascertain congestion on the road segment. It also clear that segment length has a significant impact on ID-based congestion detection, with a 24.99-meter segment length presenting the best performance result of ID-based. Additionally, the results illustrated that a longer segment length allows the ID-based congestion detection system to accurately check for vehicle density, whereas a shorter segment length limits the system's ability to detect congestion using vehicle density. Consequently, the results clearly reveal that segment length has a significant effect on congestion detection.

LETA provided a positive result. A basic method of monitoring traffic congestion in relation to vehicle destination. This provided motorists with awareness and temporal comprehension of their trip by taking into account their arrival time.

These two ways are simple to implement in the RSU of the ITS infrastructure, since it is built on the current V2X communication protocol. Both ID-based and LETA were developed to employ vehicle information, whereas ITS allows vehicles to communicate kinematic information, resulting in no communication overhead.

V. CONCLUSION AND FUTURE WORK

This study developed an ID-based congestion detection method that monitored vehicles' speed and density and loss of expected time of arrival on road segments. ID-based congestion detection method utilizes a vehicle's ID and trajectories contained in vehicular messages. Roads were divided into lengths of 8.33 meters, 16.66 meters, and 24.99 meters using vehicle trajectories. According to experimental results, the ID-based congestion detection method detects actual (true) traffic congestion with long segment lengths.

The ID-based congestion detection system combined the concept of speed threshold congestion detection with density monitoring and was applied to road segments to monitor traffic congestion. Its performance was evaluated against the speed threshold and MCDP congestion detection method. The results presented show that the ID-based congestion detection method detects congestion more accurately. Speed threshold and MCDP detect false congestion when compared with the ID-based

congestion detection method. This work shows that by using two or more congestion detection metrics, traffic congestion can be accurately detected.

This work also defines and calculates LETA by taking into account segment length, vehicle speed, and the road recommended speed. The vehicle's LETA was displayed, as well as their overall LETA for the trip. The results demonstrate that LETA can be an effective measure of traffic congestion because it displays to the motorist how much time they are losing or gaining.

This accomplishment is practical enough to be utilized in road design, moving object monitoring, and might be adapted to analyze traffic congestion in the city. To improve the ID-based congestion detection method accuracy, vehicle density monitoring must take into account the road segment length.

REFERENCES

[1] Y. Joshi, A. Joshi, N. Tayade, P. Shinde, S. Rokade, and A. Professor, "IoT Based Smart Traffic Density Alarming Indicator," *Int. Res. J. Eng. Technol.*, vol. 3, no. 10, pp. 1086–1089, 2016, Accessed: Aug. 01, 2021. [Online]. Available: www.irjet.net.

[2] J. C. Falcocchio and H. S. Levinson, "Road Traffic Congestion: A Concise Guide," *Springer Tracts Transp. Traffic*, vol. 7, pp. 159–182, 2015, doi: 10.1007/978-3-319-15165-6.

[3] ETSI, "ETSI TS 102 894-1 V1.1.1 - Intelligent Transport System (ITS); Users & Applications requirements; [Facility layer structure, functional requirements and specifications;]," 2013.

[4] C. F. Daganzo, "The cell transmission model, part II: Network traffic," *Transp. Res. Part B Methodol.*, vol. 29, no. 2, pp. 79–93, Apr. 1995, doi: 10.1016/0191-2615(94)00022-R.

[5] S. P. A. Alewijnse, K. Buchin, M. Buchin, S. Sijben, and M. A. Westenberg, "Model-Based Segmentation and Classification of Trajectories," *Algorithmica*, vol. 80, no. 8, pp. 2422–2452, 2018, doi: 10.1007/s00453-017-0329-x.

[6] G. Reyes, L. Lanzarini, W. Hasperué, and A. F. Bariviera, "GPS trajectory clustering method for decision making on intelligent transportation systems," *J. Intell. Fuzzy Syst.*, vol. 38, no. 5, pp. 5529–5535, 2020, doi: 10.3233/JIFS-179644.

[7] S. Guo, X. Li, W. K. Ching, R. Dan, W. K. Li, and Z. Zhang, "GPS trajectory data segmentation based on probabilistic logic," *Int. J. Approx. Reason.*, vol. 103, pp. 227–247, 2018, doi: 10.1016/j.ijar.2018.09.008.

[8] L. Zhu, J. R. Holden, and J. D. Gonder, "Trajectory segmentation map-matching approach for large-scale, high-resolution GPS data," *Transp. Res. Rec.*, vol. 2645, no. 1, pp. 67-75., 2017, doi: 10.3141/2645-08.

[9] M. Buchin, A. Driemel, M. van Kreveld, and V. Sacristán, "Segmenting trajectories: A framework and algorithms using spatiotemporal criteria," *J. Spat. Inf. Sci.*, vol. 3, no. 2011, pp. 33–63, 2011, doi: 10.5311/JOSIS.2011.3.66.

[10] J. Ren, Y. Chen, L. Xin, J. Shi, B. Li, and Y. Liu, "Detecting and positioning of traffic incidents via video-based analysis of traffic states in a road segment," *IET Intell. Transp. Syst.*, vol. 10, no. 6, pp. 428–437, Aug. 2016, doi: 10.1049/IET-ITS.2015.0022.

[11] M. Bani Younes and A. Boukerche, "A performance evaluation of an efficient traffic congestion detection protocol (ECODE) for intelligent transportation systems," *Ad Hoc Networks*, vol. 24, no. PA, pp. 317–336, Jan. 2015, doi: 10.1016/J.ADHOC.2014.09.005.

[12] Q. Yuan, Z. Liu, J. Li, J. Zhang, and F. Yang, "A traffic congestion detection and information dissemination scheme for urban expressways using vehicular networks," *Transp. Res. Part C Emerg. Technol.*, vol. 47, no. P2, pp. 114–127, 2014, doi: 10.1016/j.trc.2014.08.001.

[13] R. Wang, Z. Xu, X. Zhao, and J. Hu, "V2V-based method for the detection of road traffic congestion," *IET Intell. Transp. Syst.*, vol. 13, no. 5, pp. 880–885, 2019, doi: 10.1049/iet-its.2018.5177.

[14] Y. Chen, C. Chen, Q. Wu, J. Ma, G. Zhang, and J. Milton, "Spatial-temporal traffic congestion identification and correlation extraction using floating car data," *J. Intell. Transp. Syst. Technol. Planning, Oper.*, pp. 1–18, 2020, doi: 10.1080/15472450.2020.1790364.

[15] A. Annamalai, A. S. A, and R. Y. Rao, "A Review on Traffic Congestion Detection Methodologies and Tools VANET-ITS for India View project A Review on Traffic Congestion Detection Methodologies and Tools," *Int. J. Adv. Sci. Technol.*, vol. 28, no. 16, pp. 1400–1414, 2019, Accessed: Aug. 08, 2023. [Online]. Available: https://www.researchgate.net/publication/340378792.

[16] T. Afrin and N. Yodo, "A survey of road traffic congestion measures towards a sustainable and resilient transportation system," *Sustain.*, vol. 12, no. 11, pp. 1–23, 2020, doi: 10.3390/su12114660.

[17] D. Satrinia and G. A. P. Saptawati, "Traffic speed prediction from GPS data of taxi trip using support vector regression," *Proc. 2017 Int. Conf. Data Softw. Eng. ICoDSE 2017*, vol. 2018-Janua, pp. 1–6, 2018, doi: 10.1109/ICODSE.2017.8285869.

[18] R. Bauza and J. Gozalvez, "Traffic congestion detection in large-scale scenarios using vehicle-to-vehicle communications," *J. Netw. Comput. Appl.*, vol. 36, no. 5, pp. 1295–1307, 2013, doi: 10.1016/j.jnca.2012.02.007.

[19] R. Bauza, J. Gozalvez, and J. Sanchez-Soriano, "Road traffic congestion detection through cooperative Vehicle-to-Vehicle communications," *Proc. - Conf. Local Comput. Networks, LCN*, pp. 606–612, 2010, doi: 10.1109/LCN.2010.5735780.

[20] E. Zhang and X. Zhang, "Road Traffic Congestion Detecting by VANETs," in *Proceedings of the 2nd International Conference on Electrical and Electronic Engineering (EEE 2019)*, 2019, vol. 185, no. Eee, pp. 242–248, doi: 10.2991/eee-19.2019.39.

[21] Y. Shi, D. Wang, J. Tang, M. Deng, H. Liu, and B. Liu, "Detecting spatiotemporal extents of traffic congestion: a density-based moving object clustering approach," *Int. J. Geogr. Inf. Sci.*, vol. 35, no. 7, pp. 1449–1473, 2021, doi: 10.1080/13658816.2021.1905820.

[22] B. K. Chaurasia, W. S. Manjoro, and M. Dhakar, "Traffic Congestion Identification and Reduction," *Wirel. Pers. Commun.*, vol. 114, no. 2, pp. 1267–1286, Sep. 2020, doi: 10.1007/S11277-020-07420-0/METRICS.

[23] G. Yannis and E. Michelaraki, "Review of City-Wide 30 km/h Speed Limit Benefits in Europe," *Sustain. 2024, Vol. 16, Page 4382*, vol. 16, no. 11, p. 4382, May 2024, doi: 10.3390/SU16114382.

[24] G. Yannis and E. Michelaraki, "Effectiveness of 30 km/h speed limit – A literature review," *J. Safety Res.*, Nov. 2024, doi: 10.1016/J.JSR.2024.11.003.

[25] F. He, X. Yan, Y. Liu, and L. Ma, "A Traffic Congestion Assessment Method for Urban Road Networks Based on Speed Performance Index," *Procedia Eng.*, vol. 137, pp. 425–433, Jan. 2016, doi: 10.1016/j.proeng.2016.01.277.

[26] X. Yang, F. Wang, Z. Bai, F. Xun, Y. Zhang, and X. Zhao, "Deep learning-based congestion detection at urban intersections," *Sensors*, vol. 21, no. 6, pp. 1–14, Mar. 2021, doi: 10.3390/s21062052.

[27] Y. Zhang *et al.*, "A Method for Traffic Congestion Clustering Judgment Based on Grey Relational Analysis," *ISPRS Int. J. Geo-Information 2016, Vol. 5, Page 71*, vol. 5, no. 5, p. 71, May 2016, doi: 10.3390/IJGI5050071.

[28] M. A. Ahmed, A. R. M. Shariff, and S. Abubakar, "Long Term Traffic Congestion Detection Method Based on Speed-Threshold," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 15427 LNCS, pp. 25–39, 2024, doi: 10.1007/978-3-031-77003-6_3.

[29] X. Liu and Y. Ban, "Uncovering Spatio-temporal cluster patterns using massive floating car data," *ISPRS Int. J. Geo-Information*, vol. 2, no. 2, pp. 371–384, 2013, doi: 10.3390/ijgi2020371.

[30] A. Mehmood and F. Mehmood, "Vehicular Trajectories from Jeju, South Korea | IEEE DataPort," *IEEE DataPort*, 2022. https://ieee-dataport.org/documents/vehicular-trajectories-jeju-south-korea (accessed Jul. 03, 2023).

[31] M. Ahmad, Q. Chen, and Z. Khan, "Microscopic congestion detection protocol in VANETs," *J. Adv. Transp.*, vol. 2018, 2018, doi: 10.1155/2018/6387063

Stock Market Prediction of the Saudi Telecommunication Sector Using Univariate Deep Learning Models

Hadi S. AlQahtani[1], Prof. Mohammed J. Alhaddad[2], Dr. Mutasem Jarrah[3]
Department of Information Technology-Faculty of Computing and Information Technology,
King Abdulaziz University, Jeddah, Saudi Arabia[1, 2]
Department of Artificial Intelligence and Data Science, Applied Science University, Jordan[3]

***Abstract*—Stock market volatility, randomness, and complexity make accurate stock price prediction very elusive, though it is required for logical investment and risk management. This study compares four Deep Learning (DL) models, Long Short-Term Memory (LSTM), Recurrent Neural Network (RNN), Convolutional Neural Network (CNN), and a CNN-LSTM model, to predict the Saudi Telecommunication sector by focusing on the closing price time series. The daily historical closing prices of STC, Mobily, and Zain companies are gathered and preprocessed, involving duplicate removal, feature selection, and Min-Max scaling. Models were trained with MSE loss, whereas validation was done with the RMSE and MAE. The study points toward the ability of deep learning to capture complex nonlinear regression patterns in the ebbs and flows of volatile financial markets. A comparative analysis reveals that the LSTM model yielded the lowest Test RMSE in all cases (Mobily: 1.169705, STC: 0.708495, Zain: 0.27147), therefore, presenting the best overall predictive accuracy. On the other hand, RNN almost always had the highest Test RMSE values (Mobily: 1.688603, STC: 1.143664, Zain: 0.666184), highlighting its limitations. The CNN and CNN-LSTM models showed intermediate performance, with implications for enhanced financial forecasting and decision-making within this specific market segment.**

***Keywords*—Deep learning; stock market; prediction; models; regression; time series**

I. INTRODUCTION

The stock market is complex and difficult to forecast, but it is extremely valuable to financial firms, hedge funds, traders, and market speculators [1]. The share prices are influenced by numerous variables, such as economic factors, worldwide events, company reports, and market sentiment, and are therefore random and extremely difficult to predict [2]. Having the ability to forecast changes in stock prices even slightly and accurately is extremely valuable in designing investment strategies, risk management, and overall financial decision-making [4]. Historically, stock market prediction has evolved over time through various methods. Initially, many employed traditional statistical techniques such as Autoregressive Integrated Moving Average (ARIMA) models [20]. These are simple and suitable when working with stable data, but struggle when dealing with complex patterns and long-term trends often found in finance data [4]. The advancement of machine learning, particularly approaches through deep learning, has significantly transformed financial prediction. Deep learning models are distinctive in that they are able to learn from past data without depending on pre-set mathematical formulas. This is helpful in finding underlying relationships in data that linear models may not detect [9]. That is why they are particularly suitable for the fast and dynamic character of stock market data [13]. The objective of this study is to answer the question: Which deep learning architecture—RNN, LSTM, CNN, or a hybrid CNN-LSTM—provides the highest predictive accuracy and robustness in a univariate stock market forecasting? It closely examines and contrasts the predictive capabilities of individual deep learning models on the Saudi Telecommunication market. This study hypothesizes that of the selected models of deep learning (LSTM, RNN, CNN, and CNN-LSTM), the Long Short-Term Memory (LSTM) network forecasts closing stock prices of the Saudi Telecommunication industry more accurately. The reason is that LSTM is designed to learn long-term patterns in sequence data [21]. A lot of financial literature on deep learning in financial markets exists, but requires powerful and precise models on a specific stock market that forecast daily closing stock prices based on a single variable in various market scenarios. This research investigates this requirement by studying the Saudi Telecommunication sector, which is significant but not as frequently researched as major international stock indices. This study will provide useful information regarding how the new models perform in an evolving market.

This research compares various approaches in the field of stock market prediction, which has not been researched much, and it provides additional evidence. How the research is described illustrates how it was conducted, making it simple for other researchers to conduct the research again and comprehend it. It also acts as a useful guideline for additional research in this area. The research indicates that, despite being complex, deep learning models are more capable of examining stock market trends and providing a more accurate means of predicting them [24].

The remainder of this study is structured as follows: Section II examines in detail significant research regarding deep learning models applied to time series prediction. Section III describes the methods, such as how data was gathered and prepared, and how the model was constructed and trained.

Section IV presents the results of the model tests across each company. Section V discusses what these results imply, including model comparison and the study's limitations. Finally, Section VI concludes the study by outlining the major findings and providing recommendations for future research.

II. Literature Review

Deep learning models are powerful predictors of time series. They are capable of learning intricate patterns and long-term correlations without the necessity of specific feature engineering [6]. Deep models are particularly effective in detecting trends in the stock market, which usually exhibits chaotic and non-linear tendencies [13].

A. Recurrent Neural Networks (RNNs)

Recurrent neural networks tend to be appropriate when the data comes in sequence and when the position of components within it matters. RNNs have loop markings that feed-forward networks do not have. At a particular instant in time, when the output of a neuron is generated, this output is fed into the RNN as an input for the subsequent time. The name loop comes from the fact that the state in the network, called the hidden state, is retained in RNNs. They can learn patterns and relationships in sequences [1]. Hence, they can perform well over time series forecasting, as the input data is sequential, and the past step information can be recalled. The main problems of RNNs are the so-called vanishing gradient problem. When being trained, the network learns backwards; however, gradients tend to become very small with a large number of time steps, making the network difficult to learn from past data [2]. These make RNNs forget salient details from earlier steps and lack the capability to recall long sequences and times. RNNs could also suffer from overfitting, especially when considering noisy and unpredictable data such as stock prices.

The inherent recurrent structure of RNNs allows the design of a model of sequential correlations unavailable to feedforward networks [4]. Nonetheless, it is difficult for vanilla RNNs to handle longer sequences because these models have issues with exploding/vanishing gradients [14]. RNNs with memory solve this problem using gated memory cells that help with long-term dependency. The recurrent connections allow RNNs to capture temporal dependencies within the series of observations [16]. In [17], the authors discussed deep learning stock forecasting models and noted that RNNs with memory can identify temporal dependencies that are not extractable by other models.

B. Long Short-Term Memory (LSTM) Networks

LSTM came into being as an enhancement to the Recurrent Neural Network to address the vanishing gradient problem. According to research [3], LSTMs help to learn long-range patterns in sequential data. LSTMs achieve the feat because of their special inner architecture consisting of "memory cells" and three distinct "gates": the input gate, the forget gate, and the output gate [3]. These intelligent gates decide how information flows in and out through the cell state and assist the network in knowing what to remember and what to forget over time. The majority of authors who write about it assert that LSTM outperforms the basic RNN and other simplistic models in stock price forecasting. Particularly, LSTM models are mostly regarded by researchers as somewhat more accurate in stock price forecasting. Experimenting further, it has been established that the LSTM tends to make fewer errors than RNNs, especially when it comes to short-term forecasting [8].

Researchers of [10] proved that LSTM networks perform better on financial time series data characterized by nonlinearity and nonstationarity. Furthermore, they refined the algorithm to improve its predictive accuracy. In [12], the authors proposed a strong LSTM model coupled with sentiment analysis to improve stock price prediction accuracy, proving the robustness of the model in the noisy financial market. In [24], the authors confirmed that LSTM consistently outperforms RNN and CNN models in long-term prediction, thereby cementing its suitability for financial time series.

C. Convolutional Neural Networks (CNNs)

In [3], the authors noted that CNNs can be very effective at modeling short-term stock price changes, although CNNs have more trouble with long-term dependencies compared to LSTM models. A time series CNN typically contains convolutional layers. Pooling layers, such as max pooling, which reduce features to a smaller size, are followed by convolutional layers. This retains relevant features but with reduced data. Additionally, the output of these layers is converted into a 1D vector and then passed through one or more fully connected layers. These layers perform the bulk of the predicting, such as future stock prices [5].

In [7], the authors proved that CNN-based models are competitive, especially when combined with other architectures. CNNs are renowned for dealing with images and spatial information. CNNs have been used for stock market prediction, leveraging their ability to model local trends and patterns in sequential data. CNNs have also been transformed smartly to deal with 1D time series information [7]. The transformation employs convolutions on a time axis to enable CNNs to detect local patterns, features, and trends in sequential data [24].

D. Hybrid CNN and LSTM Models

The primary motivation for combining the use of both CNNs and LSTMs, is to capitalize on what each system excels at. This is a combined approach, typically beginning with a CNN scanning the raw time series data for significant, general patterns [3]. The product of these CNN layers, typically following pooling and converting it into a sequence, is then passed as input to the LSTM layers.

In [6], the authors also pointed out that hybrid models would have higher prediction accuracy at the cost of increased computational complexity. This collaboration is intended to provide a stronger insight into the time series data, including close as well as long-term connections, which typically results in more accurate prediction outcomes, as compared to the use of each model in isolation.

In [8], the authors clarified that a CNN-LSTM model performs better than single CNN and LSTM models on stock prediction tasks. Researchers of [24] proposed a hybrid CNN-LSTM model for stock forecasting and improved performance over single models by capturing both spatial patterns and temporal dependencies.

E. Comparing Various Methods of Predicting Stocks

Research papers typically compare various deep learning algorithms used in predicting stock prices, such as LSTM, RNN, and CNN ([1]-[5], [24]). The majority of these papers indicate that LSTM is typically the highest performing because it is capable of handling long-range dependencies in sequential data [5]. Other papers discussed that CNNs and RNNs are also not bad, but hybrid models such as CNN+LSTM—trained with single-price data—make minimal improvements [7]. Increasingly, more researchers are employing hybrid models to ensure predictions are accurate and dependable. According to [8], hybrid models attempt to address the issues of standalone models through the union of multiple types, frequently doing so better by blending the spatial knowledge of CNNs with the temporal considerations of RNNs or LSTMs.

III. Methodology

This study utilizes a deep learning structure, and it focuses on applying single-variable models in stock market prediction. This study entails the application and comparison of four different deep learning structures: Long Short-Term Memory (LSTM), Recurrent Neural Network (RNN), Convolutional Neural Network (CNN), and a hybrid of CNN and LSTM on the Saudi Telecommunication sector stock market.

A. Data Collection

Historical closing prices were obtained daily from the Yahoo Finance API. The source is reliable and easily accessible for financial data [15]. The companies chosen from the Saudi Telecommunication sector and their respective Yahoo Finance tickers are STC (7010.SR), Mobily (7020.SR), and Zain (7030.SR). Specifying the companies and tickers makes the dataset readable and easily reproducible. Under the single feature scheme of this study, the "Close" price per share was utilized as the only input.

B. Preprocessing Data

Accurate data preparation is paramount to the performance and strength of deep learning algorithms. The following cleaning processes were done:

1) Duplicate records were meticulously removed from the collected dataset to ensure data consistency, accuracy, and integrity.

2) As the study takes one thing at a time into consideration, only the "Close" price for each stock was kept for the model. All other potential features, such as Open, High, Low, and Volume, were excluded.

3) Min-Max Scaling is employed for normalizing the input data. It scales data linearly to a bounded range [0, 1]. Scaling enhances model convergence and numerical stability in gradient-based deep learning. The mathematical Formula (1) for Min-Max Scaling is expressed as:

$$x_scaled = \left(\frac{x - min(x)}{(max(x) - min(x)}\right) \quad (1)$$

4) Preprocessed data was divided into an 80/20 train and test split to estimate generalization and avoid overfitting.

5) Windowing/Sequence Preparation was done. We used sliding windows to create input-output pairs of data, a common practice in time series deep learning.

C. Model Building

Each company had four different deep learning models to predict stock prices using one variable. Choosing proper hyperparameters, that is, the number of layers, units (neurons) per layer, dropout rates, learning rates, batch sizes, and epochs, is very important to improve the performance of the deep learning model. The precise hyperparameter values are not given in the data, but the parameters are generally adjusted by careful methods so that the performance is optimal, facilitates convergence, and reduces problems like overfitting. Common optimizers like Adam are generally used to train such models.

D. Long Short-Term Memory (LSTM)

LSTM architecture is crafted to identify long-term trends in sequential data. It solves the common issue of vanishing gradients in standard RNNs [11]. Its building blocks therein—input gate, forget gate, output gate, and cell state—make this possible. These clever gates control the flow of information, making it possible for the network to determine what to retain or ignore through long sequences. The mathematical Formula (2) to Formula (5), governing the operations within an LSTM cell, are as follows:

$$\text{Input Gate} = i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (2)$$

$$\text{Forget Gate} = f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad (3)$$

$$\text{Output Gate} = o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad (4)$$

$$\text{Cell State Update} = c_t = f_t \times c_{t-1} + i_t \times tanh(W_c x_t + U_c h_{t-1} + b_c) \quad (5)$$

$$\text{Hidden State Update} = h_t = o_t \times tanh(c_t)$$

where,

x_t is the current input vector

h_{t-1} is the previous hidden state vector

c_{t-1} is the previous cell state vector.

W is the Weight matrix for the input

U is the Recurrent weight matrix for the hidden state

B is the bias vector

σ is the Sigmoid activation function

tanh: Hyperbolic tangent activation function.

This detailed mathematical formulation demonstrates the internal workings of the LSTM and its fundamental advantage over simpler RNNs in handling long-term dependencies, which is crucial for capturing the complex patterns in stock price movements.

E. Recurrent Neural Network (RNN)

The Recurrent Neural Network (RNN) is the basic model used to predict time series in this study. Its fundamental building block is with recurrent connections, where a neuron's output at a given time is reused as input to the network at the

next time [18]. This feedback mechanism allows the network to have a "hidden state", a memory that processes information in a sequence. The Formula (6) and Formula (7) for an RNN cell, describing how it works, are:

$$\text{Hidden state} = h_t = \sigma(W_x x_t + W_h h_{t-1} + b_h) \quad (6)$$

$$\text{Output} = y_t = \sigma(W_y h_t + b_y) \quad (7)$$

where,

xt is the current input vector

ht−1 is the previous hidden state vector

ht is the current hidden state vector

yt is the current output vector

Wx is the weight matrix for the input

Wh is the recurrent weight matrix for the hidden state

Wy is the weight matrix for the hidden state to output

bh, by are the bias vectors

σ is the activation function (typically sigmoid or tanh).

Including the RNN formula gives the explicit mathematical difference from LSTM. This explains why LSTMs were created and why LSTMs usually work better with long sequences. It shows how deep learning models have progressed in time series analysis.

F. Convolutional Neural Network (CNN)

The CNN architecture is adapted to handle time series data for the purpose of capturing local patterns and features in sequences [5]. Adaptation is mainly done through one-dimensional (1D) convolutions, which are apt for handling time data by identifying "spatial" patterns along the time axis [19]. The architecture generally comprises:

1) Convolutional layers: These layers have filters (known as kernels) that move over the 1D input data. Every filter applies a convolution operation to extract local patterns or features from subsets of the time series.

2) Pooling layers: Convolutional layers are usually followed by pooling layers, like Max Pooling, to decrease the feature maps' size. This lowers their size without losing the important features, making the model work better with small changes in input and lessening the computation involved [23].

3) Flattening and fully connected (dense) layers: The feature map output of the convolutional and pooling layers is flattened into a 1D vector. This vector is fed into one or more fully connected (dense) layers that learn useful features and carry out the final regression task [23].

4) Output layer: For predicting one type of time series, the output layer generally has one neuron to predict the future value of the stock price. The main operation of a 1D CNN is defined by the discrete convolution Formula (8):

$$(f * g)[n] = \int_{m=-\infty}^{+\infty} f[m] \cdot g[n - m] \quad (8)$$

where,

f is the input signal (the time series data)

g is the convolutional filter (kernel)

n is the position of the output element

The formula is the general definition of discrete-time convolution [22]. It continues by sliding the filter over the signal, multiplying the filter by the corresponding part of the signal at each element, and adding the outcomes to create one output point. This move reflects how methods of identifying patterns in space, normally used in images, can be used in time-based data. This shows how flexible deep learning systems are [2].

G. Combination of CNN and LSTM Model

This model is a hybrid architecture that combines synergistically the strengths of Convolutional Neural Networks (CNNs) for learning features and Long Short-Term Memory (LSTM) networks for learning sequences [3]. The architecture is made up of an early CNN component that takes in the raw time series data. This CNN block extracts important characteristics and neighborhood trends, for example, detecting some of the price changes or trends in short-time periods, from the given sequence [7]. Whatever emerges from these CNN layers, after pooling and transforming into a sequential representation, is then offered as input to the LSTM blocks. The LSTM layers then learn significant connections and timing relationships from these prepared, detailed sequences [8]. The primary reason to use this combined approach is to take advantage of the strengths of both model types: CNNs are strong at extracting powerful local patterns and reducing the size of the input [12], while LSTMs are extremely capable at interpreting time-based sequences and remembering long-term connections [12]. This is a combination to provide greater insight into the time series data, potentially providing improved predictions compared to the use of singular models [24]. The hybrid model is an ingenious method, combining the positives of each method in time series analysis, indicating the sophistication of the study.

H. Model Training

We primarily utilized Mean Squared Error (MSE) for training all the models as the loss function. We can calculate MSE with the help of Formula (9):

$$MSE = \frac{\sum_{i=1}^{n}(yi-pi)^2}{n} \quad (9)$$

where,

yi is the observed value

pi is the corresponding predicted value

n is the number of values

MSE calculates the average of the squared errors or differences, representing the average squared difference between actual and predicted values. It is typically utilized for regression problems and provides more weight to large errors compared to small errors. To verify how well the models perform on the training as well as test datasets, we utilized Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) as crucial measures.

The RMSE Formula (10) is:

$$RMSE = \left(\frac{(\Sigma(P_i - O_i)^2)}{n}\right) \quad (10)$$

where,

Pi is the value predicted

Oi is the observed value

n represents the data points.

RMSE indicates the amount of error, expressed in the same units as the target variable. All input features were normalized to the range [0,1] prior to training. The RMSE values were computed directly in the normalized space; therefore, they are unitless and represent error relative to the scaled data range. The MAE Formula (11) is:

$$MAE = \left(\frac{(\Sigma|y_i - p_i|)}{n}\right) \quad (11)$$

MAE takes the average of the difference between actual and predicted values. MAE is less sensitive to outlier values compared to RMSE and directly indicates the average magnitude of the error. In RMSE and MAE, lower values indicate good prediction and better model performance. For regression problems in time series forecasting, it is typical to use MSE as the loss function and RMSE/MAE as the measure in assessing performance. This ensures we evaluate model performance to acceptable standards in academia.

IV. Results

In this section, we display the results from the model tests of all three Saudi Telecommunication companies. The performance metrics, such as Train RMSE, Train MAE, Test RMSE, and Test MAE, for every deep learning model are presented in clear, readable tables, together with a summary, indicating key facts for each company.

Table I and Fig. 1 illustrate the performance of each of the four deep learning models on Mobily's stock price data. The findings indicate how accurate each model was in forecasting Mobily's prices, in that we can now compare the training and testing errors of the models. This is necessary for knowing how well the models can perform on new data and selecting the best model for this particular stock. For Mobily, the LSTM model recorded the smallest Test RMSE (1.169705) and Test MAE (0.913336), indicating that it was the most capable of forecasting new data out of the models. The RNN model registered the highest error rates in all metrics, indicating that it was not able to follow the exact changes in Mobily's share prices. The CNN and CNN+LSTM models performed well, with the hybrid model performing slightly better than the standalone CNN.

Table II and Fig. 2 present the performance measures of STC's stock price data for the deep learning models. For STC, LSTM performed the best with the lowest Test RMSE (0.708495) and Test MAE (0.518296), indicating that it can predict well. CNN performed the worst with the highest Test RMSE (1.784101) and MAE (1.270669), indicating that it was the most challenged by the STC data among the models. The hybrid CNN+LSTM performed better than the standalone CNN and RNN, but not as well as LSTM on Test RMSE and Test MAE.

TABLE I. Model's Performance for Mobily (7020.SR)

Model Name	Train RMSE	Train MAE	Test RMSE	Test MAE
LSTM	1.203972	0.785214	1.169705	0.913336
RNN	2.168396	1.809754	1.688603	1.334345
CNN	1.571197	1.00751	1.41664	1.115295
CNN + LSTM	1.388986	0.875195	1.388765	1.114691

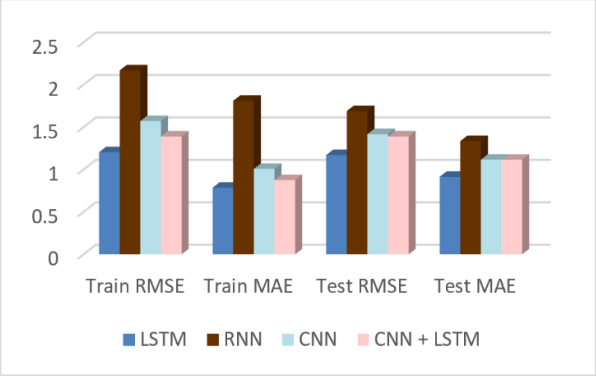


Fig. 1. Model's performance for Mobily (7020.SR).

TABLE II. Model's Performance for STC (7010.SR)

Model Name	Train RMSE	Train MAE	Test RMSE	Test MAE
LSTM	0.66395	0.367212	0.708495	0.518296
RNN	0.816863	0.494815	1.143664	0.906338
CNN	0.923335	0.533309	1.784101	1.270669
CNN + LSTM	0.780187	0.533703	1.027295	0.7784

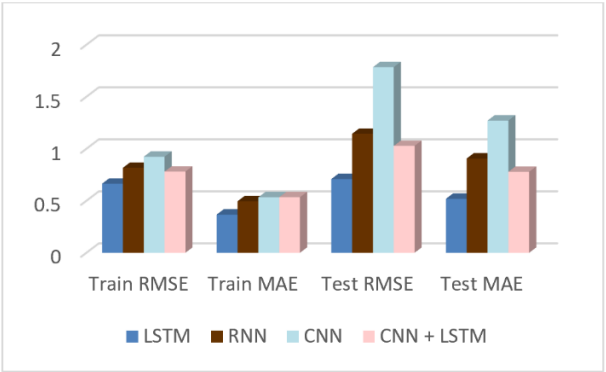


Fig. 2. Model's performance for STC (7010.SR).

TABLE III. Model's Performance for Zain (7030.SR)

	Train RMSE	Train MAE	Test RMSE	Test MAE
LSTM	0.446717	0.265711	0.27147	0.215682
RNN	0.770948	0.657144	0.666184	0.625866
CNN	0.518339	0.331256	0.302271	0.22412
CNN + LSTM	0.45294	0.281391	0.291989	0.223768

Table III and Fig. 3 present the last results for the third company. It provides the information required to make a complete comparison. It tells how the deep learning models performed with Zain's stock price data. For Zain, LSTM performed best with the smallest Test RMSE (0.27147) and Test MAE (0.215682). The CNN model also performed extremely well for Zain, placing next to the LSTM in both test metrics. The RNN model performed the worst, similar to Mobily and STC's. The CNN+LSTM hybrid model performed equally well as the pure CNN, and both are good options to forecast Zain's stock.

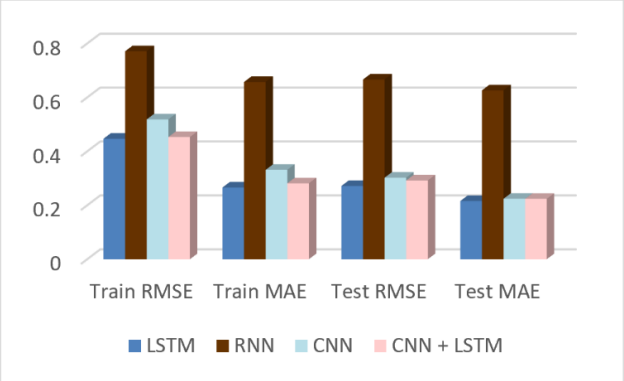


Fig. 3. Model's performance for ZAIN (7030.SR).

V. Discussion

The data in Tables I, II, and III indicate the performance of the deep learning models for three Saudi Telecommunications companies. The trend was consistent: the LSTM model always produced the lowest Test RMSE and Test MAE for Mobily, STC, and Zain, and thus was the most successful at forecasting accurate values from new data. Its consistency demonstrates the ability of LSTM to learn long-term trends and bypass the vanishing gradient problem, as cited in studies [11], and in this case, it is highly beneficial in forecasting stock prices.

A. Comparative Analysis of Models

LSTM performed stronger than the RNN models since the basic RNN model had the largest errors (MAE and RMSE) across all three firms. This is expected and is noted in other research regarding issues with simple RNNs and their lack of ability to address the vanishing gradient problem. This signifies that the models are unable to learn and remember information from long sequences [14]. This large disparity is why complex models such as LSTM were developed to address the primary issues of handling sequence data. The CNN model performed differently for each of the companies. It generally outperformed the RNN but did not always match up with the LSTM in terms of accuracy. In the case of Mobily and STC, the Test RMSE of the CNN was poorer than LSTM. Yet for Zain, the CNN model performed very competently with its Test RMSE (0.302271) and Test MAE (0.22412) close to the best of the LSTM (0.27147 and 0.215682, respectively). This indicates that CNNs are able to capture local trends and abrupt changes in the stock prices as reported in previous research ([5], [11]). The good performance for Zain can imply that its stock price variations have cleaner local trends or short-term trends that are easy for CNNs to detect. The CNN+LSTM hybrid model attempted to leverage the strength of CNN in discovering features and LSTM's ability in sequence learning. This was supposed to yield improved prediction, as various works demonstrate regarding hybrid models ([24], [3]). In the case of Mobily, the hybrid model performed slightly better than the sole CNN in Test RMSE. In the case of STC and Zain, its accuracy was comparable to or superior to single LSTM or CNN. This indicates that even if the hybrid approach is powerful, its benefits over an excellently designed single LSTM model for forecasting individual stock prices could be minor. The increased complications and computational demand of hybrid models do not always result in significantly superior performance, particularly if the data does not require both spatial discovery of features and long-term modeling of time [17].

B. Implications of Findings

It is vital to investors, traders, and decision-makers who make financial decisions in the Saudi Telecommunication market. LSTM has been performing better than any other model and has proven to be able to predict closing prices accurately every day. Identifying the best model will assist in the creation of automated trading systems or efficient forecasting tools for the market [9]. Accurate forecasts with minimal errors will provide useful information about the change in stock prices. This will assist in making more prudent investment decisions, identifying more profitable times for selling and buying, and developing more efficient risk management plans. Employing deep learning algorithms will assist in the analysis and prediction of this volatile market sector [19].

C. Limitations Addressed by this Study

This study addresses the limitations of previous research by conducting a comprehensive comparative analysis of four deep learning methods—LSTM, RNN, CNN, and a hybrid CNN+LSTM—applied to the Saudi Telecommunication sector, which encompasses three major companies and three distinct datasets. Unlike previous studies that typically rely on a single model or limited datasets, our research evaluates and compares the performance of these models across multiple datasets to enhance predictive accuracy and establish a strong basis for comparison. In doing so, this study not only identifies the most accurate predictive model for the sector but also offers practical insights into the trade-offs between model complexity and predictive performance. Moreover, by focusing on the Saudi Telecommunication market, the research fills an important geographic and industry-specific gap in the literature, providing contributions of both academic significance and practical value. Table IV summarizes the limitations of previous studies that are addressed by this study.

TABLE IV. LIMITATIONS OF PREVIOUS STUDIES ADDRESSED BY THIS STUDY

Limitations in Previous Studies	How This Study Bridges the Gap
Focus mainly on a single model (often LSTM, CNN, or LSTM-CNN).	Provides a comprehensive comparison of four deep learning models, including hybrid methods.
Use of univariate or limited datasets.	Applies models to major datasets of three companies in the Saudi telecommunications sector, giving broader validation.
Hybrids often add complexity with minimal evaluation of their true advantage.	Evaluates hybrid performance against standalone models to determine if the added complexity is justified.
Research concentrated on the US, Chinese, or Indian markets, etc.	Extends to an underexplored market (Saudi Telecom sector), contributing localized insights.

D. Limitations of the Study

This study provided useful information, but there exist certain limitations that need to be acknowledged. The model only examined the "Close" price as the input variable. This is less complicated to investigate trends in one variable, but it reduces the model's capacity to observe the complete picture of the stock market. Stock prices are influenced by various factors, such as Open, High, Low, and Volume prices, as well as external economic indicators, news sentiments, and technical indicators. Employing the multiple-variable method with the inclusion of the additional factors could result in improved and precise predictions [6]. It is greatly significant to tune the hyperparameters to achieve quality results from deep learning models [10]. These conclusions benefit the Saudi Telecommunication market but need to be examined in other markets, other contexts, or other financial markets in other nations without further tests. The unique characteristics of the market, regulation, and investors' behavior elsewhere could influence the extent to which the model works [22].

VI. CONCLUSION

This study examined the performance of four deep learning models. These models are LSTM, RNN, CNN, and a combination of CNN and LSTM. The above models were employed to predict stock prices for the Saudi Telecommunication industry. The study developed and evaluated the models using daily closing prices of Mobily, STC, and Zain. This involved meticulous data preparation and common performance-measuring methods. The overall outcome indicates that the LSTM model performed better than the other two models for all the companies. That is, it is able to learn long, non-linear, and complex patterns in stock prices. The RNN model had the largest errors, indicating that it does not handle long sequences well. The CNN model performed with mixed results, but performed exceptionally well for Zain, indicating that it is capable of identifying local patterns. The hybrid CNN+LSTM model, although a good concept, provided minor improvements over the single models in this instance. The comparison and analysis benefit the field of computational finance and deep learning in finance.

From the concepts of this research, future research should therefore consider integrating additional features such as Open, High, Low, Volume, technical indicators, and external variables (e.g., economic news, market sentiment) to enrich the predictive capacity of the models. Experimenting with advanced hybrid architectures and ensemble methods, which could better exploit the strengths of different models and capture both long-term and short-term dependencies. Expanding the application to other industries, sectors, and international markets to test the robustness and adaptability of the models under diverse financial environments.

In conclusion, this research underscores the continued dominance of LSTM in stock price forecasting, while also opening the door for refined CNN-based and hybrid approaches. By situating these results in the context of the Saudi telecommunications sector, the study provides a foundation for both academic inquiry and practical application in financial forecasting.

REFERENCES

[1] Edirisooriya, A., Weliwatta, R., Amarasena, H., & Ganesh, A. (2024). Deep learning architectures for time series forecasting. ResearchGate. https://doi.org/10.13140/RG.2.2.19904.75524

[2] Roy, S. S., R. C., Rohan C.,Kun C. L., C. S., & B M.-i. (2021), Random forest, gradient boosted machines, and deep neural network for stock price forecasting: a comparative analysis on South Korean companies, International Journal of Ad Hoc and Ubiquitous Computing 33(1), 62–71. https://doi.org/10.1504/IJAHUC.2020.104715

[3] Lu, Li, W., J., Y., Li, Wang, J. & Sun, A., (2020). A CNN-LSTM-based model to forecast stock prices. Complexity, 1–10. https://doi.org/10.1155/2020/6622927

[4] Kumar, Haider, K., & M. T. U. (2021 Application of neural network to technical analysis of stock market prediction. New Generation Computing, 39(1), 231–272. https://doi.org/10.1007/s00354-020-00104-0

[5] Borovykh, A., Bohte, S., & Oosterlee, C. W. (2021). Conditional time series forecasting with convolutional neural networks. arXiv preprint arXiv:1703.04691. https://arxiv.org/abs/1703.04691

[6] Chen, W., Hussain, W., Cauteruccio, F., & Zhang, X. (2023). Deep learning for financial time series prediction: A state-of-the-art review of standalone and hybrid models. Computer Modeling in Engineering and Sciences, 139, 187-224. https://doi.org/10.32604/cmes.2023.031388

[7] Chen, Y. C., & Huang, W. C. (2021). Constructing a stock-price forecast CNN model with gold and crude oil indicators. Applied Soft Computing, 112, 107760.

[8] Agnihotri, K., Yadav, S., Dahiya, S., Subramanyam, M., Patil, H., & Kalra, G. (2024). Next-generation stock market prediction: Integrating CNN and LSTM models. 2024 International Conference on Data Science and Network Security (ICDSNS), 1-6. https://doi.org/10.1109/ICDSNS62112.2024.10691069

[9] Khan, D. R., Patankar, A. B., & Khan, A. (2024). An experimental comparison of classic statistical techniques on univariate time series forecasting. Procedia Computer Science, 235, 2730-2740. https://doi.org/10.1016/j.procs.2024.04.257

[10] Peivandizadeh, A., Hatami, S., Nakhjavani, A., Khoshsima, L., Qazani, M. R. C., Haleem, M., & Alizadehsani, R. (2024). Stock market prediction with transductive long short-term memory and social media sentiment analysis. IEEE Access, 12, 87110–87130. https://doi.org/10.1109/access.2024.3399548

[11] García-Torres, Ayala, J., M., J. L. V., F., Divina, F., & Noguera, Gómez-Vela, (2021). Technical analysis strategy optimization using a machine learning approach in stock market indices. Knowledge-Based Systems, 225, 107119. https://doi.org/10.1016/j.knosys.2021.107119

[12] Park, K., Jayathilake, Y., Gajamannage & D. I. (2023). Real-time forecasting of time series in financial markets using sequentially trained

dual-LSTMs. Expert Systems with Applications, 223, 119879. https://doi.org/10.1016/j.eswa.2023.119879

[13] Mehtab, S., & Sen, J. (2021). A robust predictive model for stock price prediction using deep learning and natural language processing. TechRxiv. https://doi.org/10.36227/techrxiv.15023361.v1

[14] Noguer, A. M. (2024). The mathematics of recurrent neural networks. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.5001243

[15] Peng, Z. (2019). Stocks analysis and prediction using big data analytics. International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS) (pp. 309-312). IEEE.

[16] Aseeri, A. O. (2023). Effective RNN-based forecasting methodology design for improving short-term power load forecasts: Application to large-scale power-grid time series. Journal of Computational Science, 68, 101984. https://doi.org/10.1016/j.jocs.2023.101984

[17] Jiang, W. (2021). Applications of deep learning in stock market prediction: Recent progress. Expert Systems with Applications, 184, 115537. https://doi.org/10.1016/j.eswa.2021.115537

[18] Francis, M. P. Fozap (2025). Hybrid Machine Learning Models for Long-Term Stock Market Forecasting: Integrating Technical Indicators. Journal of Risk and Financial Management, 18(4), 201. https://doi.org/10.3390/jrfm18040201

[19] Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2020). Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. Applied Soft Computing, 90, 106181. https://doi.org/10.1016/j.asoc.2020.106181

[20] Siami-Namini, S., Tavakoli, N., & Namin, A. S. (2020). A comparison of ARIMA and LSTM in forecasting time series. 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), 1394–1401. https://doi.org/10.1109/ICMLA.2018.00227

[21] Vuong, P., Phu, L., Nguyen, T. H. V., Duy, L., Bao, P., & Trinh, D. (2025). A comparative study of deep learning approaches for stock price prediction. Digital Finance, 1-29. https://doi.org/10.1007/s42521-025-00149-0

[22] Yu, P., & Yan, X. (2020). Stock price prediction based on deep neural networks. Neural Computing and Applications, 32(6), 1609-1628.

[23] Zafar, A., Aamir, M., Mohd Nawi, N., Arshad, A., Riaz, S., Alruban, A., ... & Almotairi, S. (2022). A comparison of pooling methods for convolutional neural networks. Applied Sciences, 12(17), 8643.

[24] Zhao, Q., Hao, Y., & Li, Xuechen. (2024). Stock price prediction based on a hybrid CNN-LSTM model. Applied and Computational Engineering. 104. 110-115. 10.54254/2755-2721/104/20241065.

A Systematic Review of Multilingual Plagiarism Detection: Approaches and Research Challenges

Chaimaa BOUAINE, Faouzia BENABBOU, Zineb Ellaky, Amine BOUAINE, Chaimae ZAOUI
Laboratory of Artificial Intelligence and Systems, Faculty of Sciences Ben M'Sick, Hassan II University, Casablanca, Morocco

***Abstract*—The existence of voluminous multilingual sources on the web in different fields creates numerous issues, including violations of intellectual property rights. For that, the multilingual plagiarism or cross-language plagiarism detection (CLPD) has become a great challenge, which refers to copying content from a source text in one language into a target text in another without proper attribution. This study presents a systematic literature review (SLR) of methodologies used in CLPD covering works published between 2014 and 2025. This literature review summarizes and diagrams the different approaches used for CLPD. We propose a classification of the different representations of multilingual texts into four types: traditional approaches, multilingual semantic networks, fingerprinting methods, and deep learning models. In addition, we have carried out an in-depth analysis of ten language pairs, have focused on the approaches employed, including translation strategies, feature extraction approaches, classification techniques, similarity methods, dataset types, data granularity, and evaluation metrics. Among the fulfilled results, English appears in 98% of language pairs, and the English-Arabic pair stands out as the most studied. Over 60% of studies involve a translation phase with Google Translate as the most frequently used tool. The mBART model achieves over 95% accuracy for English-Spanish, English-French, and English-German, while BERT reached 96% for English-Russian. As for the assisted translation study based on the Expert translation tool, strong results are obtained for English-Persian, with an accuracy of 98.82%. On the whole, transformers offer better results in several language pairs without the need for translation.**

Keywords*—Multilingual plagiarism; systematic literature review; multilingual text representation; translation approaches; natural language processing; machine learning; deep learning

I. Introduction

Easy access to scientific and, more specifically, academic documents is a major asset for academic research. However, it is a double-edged sword, as it has led to an increase in intellectual property infringements: users do not always mention source documents. Today, in a multilingual environment, the rich existence of translation tools has further increased this problem and encouraged the growth of CLPD, where content is copied from one language and translated into another. Plagiarism is broadly defined as the unauthorized use or imitation of another person's work, ideas, or expressions, presented as if they were one's own, without attribution [1],[2]. It can take many forms, including plagiarism of ideas [3], verbatim copying and pasting [4], self-plagiarism [5], paraphrasing [6], and translation [7]. CLPD particularly refers to the act of translating content from one language to another, frequently involving paraphrasing or altering the format to mask the source [8]. There are two main types of plagiarism detection techniques: intrinsic and extrinsic. To find potential parallels, extrinsic approaches evaluate a suspect text by comparing it with information from other sources, such as databases or published documents [9],[10]. This technique works well for identifying cut-and-paste, translations, and basic paraphrases, but its effectiveness depends on the caliber and scope of the reference material used. Intrinsic methods focus on internal components such as lexical diversity, sentence structure, and writing style. Without using an external corpus, they can identify stylistic irregularities that may be signs of plagiarism [11], [12].

The CLPD system presents significantly more complex challenges than monolingual plagiarism detection due to the linguistic and cultural diversity between source and target texts. Firstly, the language barrier makes it difficult to directly identify similarities when texts are written in different languages [13]. Secondly, reliance on machine translation systems can introduce semantic or syntactic errors and compromises the reliability of post-translation comparisons [14]. Moreover, authors intending to conceal plagiarism may reformulate ideas using stylistic variations [15] and paraphrasing specific to the target language [16], making it even harder to detect correspondences. Finally, in many cases, there are no aligned parallel resources or sufficient bilingual corpora for certain language pairs, which limits the effectiveness of approaches based on comparable or aligned corpora [17].

To meet these challenges, CLPD relies on several main approaches. The first is to use machine translation to convert documents into a common language, and then apply traditional text-matching methods [18]. Although simple, this strategy is highly dependent on the quality of the translation. A second approach avoids direct translation by using multilingual embedding models that project texts from different languages into a shared vector space, enabling direct semantic comparison [19]. Another method uses knowledge graphs, where multilingual semantic networks such as WordNet are employed to represent conceptual relationships between terms [20]. Finally, the fingerprinting technique extracts distinctive features (such as n-grams) from texts and compares them across languages [21]. These approaches make it possible to identify semantic similarities without relying solely on word-for-word translation.

This study aims to conduct an SLR of CLPD. To the best of our knowledge, only one article has addressed multilingual text plagiarism detection [22]. It is crucial to conduct a review of the existing reviews on CLPD because the field of CLPD detection is constantly evolving, and new methods and techniques are being developed. This SLR covers research on CLPD techniques published between 2014 and 2025 and addresses

various Multilingual Text Representation Strategies. This SLR study offers a concise and clear summary of the methodology currently used in CLPD. Our work provides a comprehensive analysis of all phases involved in CLPD, including data preprocessing, feature extraction, models used, similarity measures, and dataset utilization. Additionally, we extracted a taxonomy of methodologies employed in CLPD, a level of methodological integration and synthesis that had not been thoroughly addressed in prior state-of-the-art research.

The significant contributions of this research work are as follows:

- Comprehensive review of most studies addressing CLPD.
- Proposition of a systematic classification of multilingual text representations into four main categories: traditional approaches, multilingual semantic networks, fingerprinting methods, and deep learning models.
- In-depth analysis of ten language pairs, emphasizing the approaches employed, including translation strategies, feature extraction methods, classification techniques, similarity measures, dataset types, data granularity, and evaluation metrics.
- Detailed examination of detection techniques specific to each language pair.
- Proposition of an architectural framework for CLPD, integrating the insights gained from the literature and the conducted analyses.

This research study is structured as follows: Section II presents the related work. Section III presents the systematic review methodology. Section IV offers an overview of the state-of-the-art techniques in CLPD, examining the considered language pairs, feature extraction strategies, datasets, and model performances. We present the analysis of the research questions and structure the analysis into two distinct parts for better clarity and understanding. The first part provides a general overview and examines all the approaches used for CLPD. The second part focuses on a detailed and specific analysis of each language pair. In Section V, we generate the architectures employed in CLPD methods and results. And, we highlight promising research directions that remain largely unexplored. Finally, we summarize the main results and shed light on future work in Section VI.

II. Related Work

This section aims to present a summary of studies on CLPD approaches, exploring various approaches and techniques developed between 2014 and 2025. We analyze the studied language pairs, feature extraction strategies, datasets, and model performances. This review highlights the progress made through the integration of advanced models, such as transformers, and the use of multilingual resources while identifying persistent limitations, particularly for under-resourced languages. To structure this analysis, we classify existing works into four main categories based on the feature extraction step: 1) traditional approaches, encompassing classical word embedding techniques such as Word2Vec, TF-IDF, as well as statistical methods like LSI (Latent Semantic Indexing) and SVD, which rely on algebraic transformations; 2) transformers & deep learning, including modern neural-based models such as transformers like BERT, XLM-R, as well as deep learning architectures like GRU, which learn contextualized text representations through neural networks; 3) approaches based on Multilingual Semantic Networks (MSN) such as WordNet, BabelNet, which rely on predefined linguistic knowledge rather than statistical or neural-based learning, often combined with similarity measures such as Wu-Palmer (WuP), Lin, and Jaccard for Cross-Lingual (CL) text similarity assessment; and 4) fingerprinting techniques, which use text fingerprinting techniques like Winnowing, N-grams, and hash-based techniques to capture distinct textual patterns and structures for plagiarism detection and text similarity assessment. A systematic comparison of previous research is easier due to this classification, which clearly distinguishes various methods according to how they represent text and extract features.

A. Traditional Approaches

The authors of this paper [23] proposed a method named CL-WE-Tw, which combines word2vec with part-of-speech (POS) features, the Term Frequency-Inverse Document Frequency (TF-IDF) weighting technique, and MUSE for the Arabic-English (Ar-En) language pair. The proposed model achieved a Pearson correlation (PC) of 81.47% on the SemEval-2017 dataset. In [24], the authors presented an approach for Ar-En CLPD using several techniques, including CL Conceptual Thesaurus-based Similarity Continuous Bag-of-Words (CL-CTS-CBOW), CL Word Embedding Similarity (CL-WES), and Inverse Document Frequency (IDF). The IDF method achieved an F-score of 88% at the word level and 82.75% at the sentence level on four datasets (Books, Wikipedia, EAPCOUNT, and MultiUN). A method to measure semantic textual similarity for Spanish and English (Es-En) sentences was proposed in [25] to detect CLPD. The techniques include CL-CnG (character n-grams), CL-CTS (conceptual thesaurus), CL-WES (word embeddings), and T+WA (word alignment after translation), as well as supervised and unsupervised combinations, notably using the M5' model. The method achieved a correlation of 88.02% on SemEval-2016 and 83.02% on the SNLI corpus of SemEval-2017. In [26], the authors proposed the CL Word Mover's Distance (WMD) method and evaluated similarity in an aligned multilingual space for a Chinese-English text. Vector word representations are formed independently for each language using the Skip-Gram model and aligned using a small bilingual dictionary. On the NDLTD dataset, CL-WMD achieved a Hit@10 of 97.09% at the paragraph level and 86.09% at the sentence level. In [27], a CLPD was developed for the English and Russian languages. The method translates Russian documents into English using a Transformer-based machine translation system and applies semantic clustering with FastText anchors for source retrieval and unsupervised and semi-supervised sentence anchors for document comparison. Using a dataset synthesized from Russian and English Wikipedia, the system achieved an F1 score of 80%. On the PAN'11 dataset, the system achieved a precision of 94%, a recall of 76%, an F1 score of 84%, and a PlagDet score of 83% for monolingual plagiarism detection. The article [28] focused on detecting English-Arabic (En-Ar) using methods to extract semantic and syntactic features, such as word order, word embeddings (Word2Vec),

TF-IDF, and word alignments with multilingual encoders MUSE. These techniques, combined with the different ML algorithms, including Support Vector Classifier (SVC), Logistic Regression (LR), Linear SVC, DT, KNN, and Extreme Gradient Boost (XGBoost), were used to determine whether the sentences were plagiarized. The approach achieved an F1-score of 0.879 on the SemEval-2017 dataset, with the SVC classifier by integrating all the proposed techniques. Authors of this paper [29] suggested a CLPD technique for English-French (En-Fr) utilizing word embeddings generated through the CBOW model alongside Cosine Similarity (CS) to assess both semantic and syntactic similarities. Among the methods employed are CL Word Embedding Similarity (CL-WES), which evaluates Sentence Embeddings (SE) directly, syntactically enhanced embeddings (CL-WESS), and integration strategies such as decision tree-based methods. The evaluation was conducted on a comprehensive dataset combining texts from Wikipedia, conference papers, product reviews, Europarl, and JRC. The approach achieved an F1 score of 89.15% at the chunk level and 88.5% at the sentence level using decision tree fusion. The authors of this article [30] proposed to detect plagiarism between Ar-En pairs using a Semantic Textual Similarity (STS) and Skip-Gram and CBOW embedding techniques. Three learning approaches were implemented: Parallel Mode, Word by Word Alignment Mode, and Random Shuffling Mode. The combination of the Random Shuffling method and the Skip-Gram technique achieved an important performance, with a correlation rate of 75.7% on the SemEval-2017 dataset. In [31], the authors introduced a CLPD approach based on multilingual word embeddings to align plagiarism fragments across languages. The method focused on the German-English (De-En) and Es-En language pairs and is evaluated on the PAN-PC-11 and PAN-PC-12 datasets. For the candidate retrieval phase, potential fragments are identified using a vector space model with TF-IDF-like weighting and CS. For the detailed analysis phase, a word-graph representation is used to capture semantic and syntactic relationships through clique-based graph matching. The method achieved PlagDet scores of 85.7 (De-En) and 83.5 (Es-En) on PAN-PC-11, and 86.2 (De-En) and 84.2 (Es-En) on PAN-PC-12. Additionally, the method achieved a PC of 44.3 on the SemEval 2017 dataset for Es-En. The authors of [32] focused on the candidate retrieval phase for the De-En language pair, aiming to identify potential source documents for suspicious texts. It uses an approach based on thematic segmentation, Latent Dirichlet Allocation (LDA), proximity-based language models, and the extraction of keywords using TF-IDF, along with representative phrases (bigrams and trigrams). Representative words and phrases are translated using Google Translate, reducing reliance on full document translation. Experiments on the PAN-PC-12 corpus show an F2-score of 67.03% by combining thematic and linguistic segmentation. This article [33] focused on candidate document retrieval to effectively identify potentially plagiarized sections between Chinese and English texts. The developed techniques included methods based on keywords using TF-IDF and machine translation, utilizing tools such as BABYLON and Google Translate. Performance results show that queries based on 50% of the keywords provide a balance between efficiency and accuracy (MAP = 0.500) when applied to the Xinhua Chinese and English news collections. Authors of [34] presented an approach based on two steps: candidate fragment identification and detailed analysis. In the first stage, the aim is to identify potentially plagiarized fragments. It begins by selecting a subset of words, termed representative terms, which characterize the vocabulary used in the source document using the TF-IDF method. This approach does not require a complete translation of the text; only the representative terms are translated. Several translation resources were evaluated for this purpose, including Google Translation, BabelNet, and Dict.cc. Among these resources, Google Translation achieved the PlagDet score, reaching 75.42% similarity, with a precision of 72.67%. The second stage involves aligning the source and suspicious fragments, followed by testing their similarity using dynamic programming algorithms. Different term weighting models were utilized, including the TF-IDF model, the Bernoulli weighting model, the Bose-Einstein weighting model, the Simple IDF model, and the Binary model. The TF-IDF model achieved a plagdet of 91.66%, the Bernoulli weighting model reached a plagdet of 91.21%, the Bose-Einstein weighting model obtained a plagdet of 91.88%, the Simple IDF model reached a plagdet of 91.82%, and finally, the Binary model achieved a plagdet of 92.08% specifically for Es-En text pairs. In this article [35], the authors developed two approaches for the candidate retrieval task in CLPD methods. In the first approach, the most representative words for search queries are extracted using the TF-IDF technique, while in the second, the concepts are extracted and documents are semantically represented using the CL Explicit Semantic Analysis (ESA) method. The CL-ESA model was developed using Wikipedia by utilizing interlingual relationships between English, German, and Spanish articles. This allowed content to be mapped into a common conceptual space to handle problems such as polysemy and synonymy. These two methods were merged to create a hybrid model that uses dynamic interpolation to integrate similarity scores based on concepts and keywords. Evaluated on datasets such as PAN-PC-12, JRC-Acquis, PAN-PC-11, and Wikipedia, the hybrid model achieved a recall of 78.89%, a precision of 61.74%, an F1 score of 69.27%, and an F2 score of 74.74. In [36], the authors proposed a CLPD method for the English-Persian language pair, and five methods were used, each with distinct techniques and precision levels. The Translation plus Mono-lingual Analysis (T+MA) method, combining machine translation and mono-lingual analysis, achieved the highest precision of 83%. The Bilingual Word Embeddings Without Alignment (BILBOWA) method, using bilingual word embeddings, reached a precision of 55%. CL Latent Semantic Indexing (CL-LSI) created a multilingual semantic space and achieved 49% precision, while CL-ESA, based on similarity to Wikipedia documents, resulted in a lower precision of 21%. Lastly, CL Knowledge Graph Analysis (CL-KGA), utilizing BabelNet for knowledge graph analysis, showed an intermediate precision of 70%. The article [37] proposed a method for detecting bilingual plagiarism for En-Pe documents. Based on the Vector Space Model (VSM), the technique utilizes morphological analysis, synonym lists, and bilingual dictionaries to compare textual content. The approach relies on term weighting with TF-IDF to assign a numerical weight to each word. The Text similarity is then measured using CS. The dataset includes 100 training texts and 100 test texts extracted from internet sources. The performance is evaluated with 88% precision, 96% recall, and an F-measure of 91%. In

[38], the authors addressed the identification of Vietnamese-English paraphrases using Siamese recurrent architectures, incorporating techniques such as Siamese Long Short-Term Memory (SLSTM), Word2Vec-based bilingual word embedding mapping, adding POS vectors to word embeddings, and revising POS labeling tags using WordNet and VietNet. The proposed method achieved an accuracy of 89.61% on the TED dataset. In this article [39], the system proposed aims to assess semantic similarities between Indonesian and English documents by combining Latent Semantic Analysis (LSA), Singular Value Decomposition (SVD), and Learning Vector Quantization (LVQ) for classification. The system utilizes a local dictionary database for word-by-word translation, ignoring grammatical rules, and calculates similarities through CS using the Slice and Pad methods as well as the Frobenius norm. When tested on manually translated scientific articles, the system achieved a maximum accuracy of 87%. Furthermore, using a term-document matrix based on term frequency significantly outperforms a binary method in terms of accuracy. The authors in [40] addressed the challenge of En-Ar texts by using latent semantic indexing (LSI) for both English and Arabic texts. Latent semantic indexing constructs a shared semantic space for both languages, enabling contextual similarity comparisons between documents without the need for direct translation. When utilized on the parallel En-Ar Corpus of United Nations documents (EAPCOUNT), the LSI technique resulted in a similarity detection rate of 93%, while Jaccard's index yielded just 33%. The authors of this article [41] developed a CLPD system that examines both syntactic and contextual similarities among texts in various languages, such as English-Chinese, English-Japanese, and English-Korean. The system employs sophisticated methods, including Word2Vec and a model based on Convolutional Neural Networks (RCNN). It achieved commendable accuracy on the student dataset, reporting 88.87% for English-Chinese, 87.49% for English-Japanese, and 87.26% for English-Korean. In this article [42], a system is proposed for detecting text reuse for the English-Urdu languages through the development of the CLEU corpus. The goal is to create a realistic resource for detecting multilingual plagiarism. The method used, Translation Plus Monolingual Analysis (T+MA), first translate the Urdu text into English using Google Translate, before applying similarity techniques such as n-gram overlap, Longest Common Subsequence (LCS), and Greedy String Tiling (GST). Pairs of sentences/passages are classified as Near Copy, Paraphrased Copy, and Independently Written. The results show an F1 score of 73.2% in binary classification and 55.2% in ternary classification.

B. Multilingual Semantic Network

A Multilingual Semantic Network is a knowledge base that represents semantic relationships between words or concepts across multiple languages. It captures the meanings and relationships between terms in different languages, enabling tasks such as machine translation, multilingual information retrieval, or natural language understanding (NLU) in a multilingual context. These semantic networks are often built by linking words or concepts across languages through lexical or conceptual alignments, using resources like WordNet [43], BabelNet [44], or other multilingual lexical databases.

Fuzzy semantic similarity techniques were developed in [45] for Ar-En within a Big Data framework. Performance metrics were based on the Wu&Palmer and Lin similarity measures, leveraging WordNet to assess semantic relationships between words. Experiments conducted in a Hadoop environment (HDFS and MapReduce) demonstrated that the Wu&Palmer measure achieved a precision of 54%, recall of 66%, and F-measure of 59.4%, while the Lin method achieved a precision of 27%, recall of 37%, and F-measure of 31.2% on a dataset comprising news, articles, tweets, and academic works. The work in [46] addressed CLPD for the Urdu-English language pair using ML models. Using Jaccard and CS were calculated for uni-grams and tri-grams, with lemmatization performed using WordNet during preprocessing. Five classifiers were utilized: Naïve Bayes (NB), Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Decision Tree (DT), and Random Forest (RF). Among them, KNN achieved an accuracy of 92%. In the article [47], the authors proposed the CL Ontology-Based Similarity Analysis (CL-OSA) model, focusing on multilingual text similarity for languages such as Es-En, De-En, Chinese-Japanese, and French-English. The CL-OSA model leverages Wikidata, a multilingual open knowledge graph, to represent documents as language-independent entity vectors. CL-OSA applies a semantic hierarchy-based weighting and CS to rank document similarities. Evaluation of datasets such as PAN-PC-11, ASPEC-JE, ASPEC-JC, JRC-Acquis, and Europarl. For candidate retrieval, CL-OSA achieved a Mean Reciprocal Rank (MRR) of 91.38% for Es-En, 71.92% for Japanese-English, 78.21% for Japanese-Chinese, 97.68% for French-English, and 55.47% for French-English. For detailed analysis, CL-OSA achieved a PlagDet score of 57.3% for Es-En and 52.1% for De-En. The authors in [48] proposed a Multi-Lingual Plagiarism Detection (MLPD) model for English-Persian texts, focusing on translational plagiarism. The method utilizes translation tools, including manual translations by English experts and Google Translate, to translate Persian texts into English, followed by semantic similarity evaluation using the WUP metric. In the candidate retrieval phase, the system uses Apache Solr to identify the five most probable source documents by calculating the frequency of suspicious phrases in the indexed dataset. The findings highlight the significance of accurate translations on the Mizan dataset, with the suggested method achieving an accuracy of 98.82% with expert translations and 56.9% with Google Translate. A CL Knowledge Graph Analysis (CL-KGA) method was used in [49], which uses BabelNet to construct knowledge graphs and measure semantic similarity between document fragments. The languages studied are De-En and Es-En. The performances show that for automatic translations (AT), the PlagDet score reaches 60.87% for the Es-En pair and 52.96% for the De-En pair, while paraphrastic translations (PT) reach scores of 9.93% and 10.06%, respectively. The similarity measures are based on intersection algorithms and the weighting of concepts and graphical relationships. In [50], the authors proposed a method for detecting similarities between Arabic and English documents using Linear LR for classification. The approach starts with key phrase extraction and the translation of Arabic documents into English, followed by similarity detection using techniques like CS, LCS, and N-grams (tri-grams), with the integration of synonyms through WordNet. Tested on Wikipedia articles, the method achieved a precision of 96%, a

recall of 85%, and an F-measure of 90%. This work [51] proposed a fuzzy approach to inter-language (French-Arabic, En-Ar) plagiarism detection by combining semantic similarity with an Apache Hadoop-based Big Data environment. Documents are pre-processed (tokenization, empty word removal, POS tagging, and trigram segmentation) and analyzed using WordNet and similarity measures such as WuP, Lin, and Leacock-Chodorow (LCH). A test corpus of 600 documents was used, including 200 automatically translated and 400 translated with modifications (paraphrasing, back-translation). The results show that WuP delivers the best performance, with a similarity percentage of around 63%. In [52], the authors addressed two tasks: CL Text Semantic Similarity (CL-STS) and plagiarism detection (PD) for En-Ar texts. It relies on several semantic features, including topic similarity, Named Entity Recognition (NER), semantic role labeling (SRL), spatial role labeling (SpRL), stop word bag, and meaning bag. Topic generation is performed using LDA, complemented by the use of BabelNet to extract English synonyms from Arabic topics and the WUP metric to assess semantic similarity. Arabic texts are translated into English using the Google Translate API to enable analysis in a common language. For plagiarism detection, three models were used: Deep Neural Networks (DNN), LR, and SVM. The results showed that the SVM model achieved an accuracy of 96.65%, the LR model 96.64%, and the DNN model 97.01%.

C. Fingerprints

In this study [53], the authors developed a method for detecting CLPD for En-Ar, utilizing techniques such as keyphrase extraction to compute phrase frequency and rank candidate keyphrases, along with C-Value and NC-Value algorithms, translation of keyphrases into English, and fingerprinting for document representation. Five similarity measures were employed: N-Grams Similarity, LCS, Dice Coefficient (DC), Jaccard Similarity, and Containment Similarity, which were used as features to train three ML models: SVM, NB, and LLR. The results demonstrated that SVM achieved an F-score of 92% when more than three similarity measures were combined. This paper [54] proposed a method for Chinese-English based on WordNet. The approach extracts nouns using ICTCLAS for Chinese and the Stanford POS Tagger for English, encoding them into language-independent fingerprints via WordNet's semantic hierarchy. A disambiguation algorithm based on semantic density calculates the relevance of senses to address word polysemy. The method measures similarity using the Dice coefficient to compare fingerprints and identifies potential plagiarism cases when the similarity exceeds a predefined threshold. Tested on a parallel corpus, National Knowledge Infrastructure (CNKI), the proposed method achieved a precision of 87% and a recall of 78%. In [55], the authors developed a plagiarism detection system for multilingual documents (English-Indonesian) using the Winnowing method and the Jaccard coefficient to measure similarity. The system allows users to upload documents in text or PDF format, translate them automatically, if necessary, pre-process them (folding capital letters, tokenization, removal of empty words), and generate fingerprints for comparison. The dataset used includes academic journals from sources such as IEEE Xplore and ResearchGate. The system achieved an accuracy of 84.7%. The authors of [56] proposed for the Ar-En language pair, the method utilizes the Winnowing algorithm to generate digital fingerprints from k-grams extracted from documents, achieving 81% recall, 97% precision, and an 89% F-measure on Wikipedia articles.

D. Transformers & Deep Learning

The pre-trained models Multilingual BERT (M-BERT) and CL Roberta (XLM-R) [57] were used for the English-Vietnamese language pair. For this task, XLM-R achieved an accuracy of 84.3% and an F1-score of 87.6%, while M-BERT achieved an accuracy of 73.7% and an F1-score of 81.3% on the GLUE dataset. An AraXLM method based on the XLM-RoBERTa model was proposed in [58] to detect plagiarism for Ar-En texts. The framework utilizes the SemEval-2017 Task1 dataset, where the Arabic sentences were translated from English using Google Translate. The approach includes automatic diacritization, semantic similarity calculation using FAISS (Facebook AI Similarity Search), and measures such as CS and Euclidean Distance (ED) to compare the embeddings of sentences. In testing, CS achieved 94.49%, and ED was measured at 1.74668 for non-diacritized sentences. This paper [59] proposed an mBART transformer for feature extraction combined with SLSTM. Experiments were conducted on language pairs including En-Fr, En-Es, and En-De. The methodology achieved an accuracy of 98.83% and an F1-score of 98.87% for En-Fr, 97.94% accuracy and an F1-score of 98.01% for En-Es, and 95.59% accuracy with an F1-score of 96.02% for En-De. The evaluation was performed on datasets such as PAN-11, JRC-Acquis, Europarl, Wikipedia, and conference papers. This work [60] explored various methods for addressing Russian-English text alignment in the context of plagiarism detection. The method translated the Russian text into English using Neural Machine Translation (NMT). To identify translated plagiarism, a comparative analysis of models including SE, BERT, Word Substitution (WS), and LASER was conducted. With an F-score of 95%, a precision of 96%, and a recall of 93%, the BERT model stood out among the others. The study also assessed the similarity of translated sentences using Jaccard metrics with 1-grams (NMT) and 2-grams (NMT2). The NMT model demonstrated a precision of 85%, a recall of 80%, and an F-score of 82%. Furthermore, LR was applied in two configurations: LR-1, which incorporated all techniques, and LR-2, which focused on SE and WS. LR-1 attained a precision of 91%, a recall of 80%, and an F-score of 85%. In [61], the authors addressed the case of plagiarism in the Persian-English pair using multilingual transformers models (XLM-R, M-Bert, DistilBert) and the CS metric. The findings of the study indicate that the XLM-RoBERTa model achieved a PC of 95.62% on the PESTS dataset. In comparison, M-BERT achieved a correlation of 91.88%, while DistilBERT achieved a correlation of 89.51%. The authors of [62] proposed a technique that combines data augmentation with a vigilant Siamese LSTM model to detect plagiarism between Tibetan and Chinese. For Tibetan-Chinese sentence pairings produced from the CWMT and SICK datasets, this method produced a PC of 54.76%. The similarity between document abstracts was also assessed using a Doc2Vec model, yielding Tibetan-Chinese PC scores of 87.06% for Chinese, 63.67% for Tibetan, and 53.9% for Tibetan-Chinese. This work [63] proposed a CLPD approach using a graph transformer-based model (CL-GTA) and knowledge graphs (KG). After creating KG from the Extended Open Multilingual WordNet, the model employed a graph transformer to weight entities and

semantic relations using a multi-head attention mechanism. CS was applied to measure text similarity, and the system was evaluated on multiple datasets, including Europarl, JRC-Acquis, Wikipedia, PAN 2011, and Ar-En parallel corpora. The CL-GTA model achieved a PlagDet of 62% for Es-En, 58.4% for French-English, and 52.2% for Ar-En**.** In [64], the authors proposed Doc2Vec+SLSTM CLPD in the Es-En pair combines Doc2Vec for text representation and an SLSTM model to evaluate document similarity. The performances were compared to other techniques, such as GloVe, FastText, BERT, Word2Vec, and Sent2Vec, using datasets from PAN11, JRC-Acquis, Europarl, and Wikipedia. The Doc2Vec+SLSTM model achieved the highest accuracy of 99.81%, outperforming GloVe (99.59%), FastText (98.82%), BERT (99.49%), Word2Vec (99.14%), and Sent2Vec (98.41%). This paper [65] identified paraphrases for English and Vietnamese sentences using a hybrid approach. The method integrates a Fuzzy-based approach leveraging BabelNet to evaluate semantic relationships between words, a Siamese LSTM model to calculate sentence similarities, and feature combination methods employing algorithms such as LR, RF, and Multilayer Perceptron (MP). BabelNet, enhanced with VietNet, addressed its limitations by adding more Vietnamese words to the synsets. The model was evaluated on a bilingual English-Vietnamese corpus derived from TED. The proposed approach with Linear Regression yields a precision of 80.3%, a recall of 95.8%, and an F-measure of 87.4%. In [66], the authors evaluated the effectiveness of six multilingual transformer-based models for CLPD. The models tested are mBERT, mDistilBERT, XLM-RoBERTa, SBERT Multilingual MiniLM-L12, SBERT Multilingual MPNet, and Distil SBERT Multilingual. The study was conducted on language pairs formed between English and ten languages from the Indo-European family: En-Es, En-Pt (Portuguese), En-Fr, En-Ru, En-Sr (Serbian), En-Cs (Czech), En-De, En-Nl (Dutch), En-Sv (Swedish), and En-Hy (Armenian). The datasets used include sentences from Wikipedia translations to generate positive examples and test sets derived from the Microsoft Research Paraphrase Corpus (MRPC), as well as specific sets like Negative-1 and Negative-4 for the En-Ru pair. The authors of [67] presented an approach focused on the English-Urdu language pair. It leverages recent multilingual models such as LLaMA and Mistral to extract shared semantic representations across languages. These representations are then used to compare content between different languages. The evaluation was conducted on the CLPD-UE-19 dataset, and the results report an F1 score of 73.9%.

III. Research Methodology

A systematic literature review is a structured method of reviewing academic literature that involves collecting and critically evaluating a set of articles focused on a specific topic. Its purpose is to identify, select, synthesize, and analyze relationships, limitations, and key findings, thereby providing a comprehensive summary of both quantitative and qualitative studies [68]. Conducting an SLR is essential for researchers to explore current research trends related to the detection techniques of Multilingual Plagiarism, and to identify weaknesses in existing methodologies.

A. Search Strategy

To ensure the relevance and comprehensiveness of the studies retrieved, we make use of the most widely recognized research databases, as the choice of search strategy significantly influences the quality and completeness of the results. The databases included Scopus, ScienceDirect, IEEE Xplore Digital Library, SpringerLink, ACM Digital Library, ResearchGate, Web of Science, arXiv, and Google Scholar. Additionally, the Rabbit tool is employed to enhance the efficiency of the search process and to enable the identification and organization of relevant studies more effectively.

B. Research Questions

To gain a comprehensive understanding of the issue of CLPD along with its associated challenges, we formulate the research questions that this SLR aims to address as follows:

RQ1: Which language pairs are most commonly studied in CLPD, and which languages dominate research?

RQ2: What are the common translation strategies used in CLPD systems, and what are the main types of approaches adopted in this field?

RQ3: What are the different multilingual representation methods used in the feature extraction step, and what is their impact in terms of performance?

RQ4: What similarity measures are used in CLPD, and which one is the most commonly dominant in the existing studies?

RQ5: What are the main factors (preprocessing, datasets, multilingual semantic networks, feature extraction techniques, ML/DL models, Similarity Measures) that influence performance in each language pair?

C. Query Terms

Our research utilizes a "snowball" method to identify relevant literature in the field of CLPD, and starts the literature review with a few critical or foundational selected studies. Based on the references included in these selected studies, additional publications are found. This process is carried out repeatedly. The approach persists until no additional pertinent studies are discovered. The detailed steps of our snowball search method are outlined below.

- Applying text mining and natural language processing to an initial set of CLPD studies enabled the extraction of key terms, including concepts like cross-language alignment, semantic similarity, and translation-based detection.
- Constructed from these terms, a search query using relevant keywords and Boolean operators is formulated to ensure effective retrieval.
- Conducted across major databases such as IEEE Xplore, SpringerLink, ScienceDirect, ACM Digital Library, and Google Scholar, the search aimed to capture both breadth and depth.

- Manually reviewed, the results were categorized as Relevant (R) or Not Relevant (NR) based on their alignment with CLPD objectives.
- Refined through further analysis, the keyword set is adjusted with generic terms removed and domain-specific ones like “semantic similarity” or “sentence alignment” included.
- Extended by snowballing, each relevant paper’s references and subsequent citations are explored to trace both foundational and recent contributions.
- Repeated iteratively, the process concludes when no additional relevant studies emerge, resulting in a curated set of literature for the CLPD review.

Query 1: ("cross-language plagiarism" OR "cross-lingual plagiarism" OR "multilingual plagiarism detection" OR "cross-language text reuse" OR "translation plagiarism").

Query 2: ("plagiarism detection" OR "copy detection" OR "text similarity detection" OR "text matching" OR "semantic similarity").

Query 3: ("multilingual embeddings" OR "word embedding" OR "sentence embedding" OR "machine learning" OR "deep learning" OR "transformers").

Query 4: ("translation-based" OR "translation independent" OR "machine translation" OR "neural translation").

Query 5: ("systematic literature review" OR "SLR" OR "survey" OR "comparative analysis" OR "performance evaluation").

D. Study Selection

To ensure the transparency and scientific rigor of our SLR, we defined explicit inclusion and exclusion criteria to guide the selection process. After retrieving the initial set of studies through our search queries, a two-stage screening was conducted. In the first stage, titles and abstracts were reviewed to quickly discard irrelevant works. In the second stage, the full texts of the remaining studies were assessed against the predefined criteria. Only those studies that satisfied all inclusion criteria and none of the exclusion conditions were retained. This systematic filtering ensured that the final corpus focused exclusively on peer-reviewed, English-language research directly addressing CLPD with robust methodological contributions. The complete list of inclusion and exclusion criteria is presented in Table I.

TABLE I. LIST OF INCLUSION AND EXCLUSION CRITERIA

Inclusion Criteria	Exclusion Criteria
Articles between 2014 and 2025	No experimental methods are employed.
The papers are written in English	Papers unrelated to the research questions defined in our SLR.
Scientific papers are published through conferences or journals.	Closed access or missing full text.
The study focuses on CLPD	Research papers use monolingual plagiarism
The articles apply one or more of the following methods: MSN, feature extraction, machine learning, or deep learning.	Preprint papers

E. Quality Assessment

The quality assessment stage in an SLR aims to rigorously evaluate the methodological validity, reliability, and overall relevance of the studies selected [69]. This step acts as a decisive filter, applied to the full-text articles, and marks the final stage in preparing the dataset for data extraction and synthesis. To ensure the inclusion of only high-quality and meaningful contributions, a multi-phase evaluation process is employed [70]. The first phase consists of a preliminary screening of the title, abstract, conclusion, and keywords to eliminate clearly irrelevant studies. In the second phase, the remaining papers undergo a thorough assessment based on predefined quality criteria, such as methodological rigor, clarity of results, and alignment with the research questions. Only the studies that satisfy these standards are included in the final synthesis. The quality assessment rules are defined in Table II:

TABLE II. LIST OF QUALITY ASSESSMENT QUESTIONS

ID	Quality Assessment Questions
QA1	Does the article explicitly address one or more of our research questions?
QA2	Is the scope of the research clearly outlined and focused on CLPD?
QA3	Is the contribution of the study to CLPD clearly stated?
QA4	Is the contribution well-supported by evidence or evaluation?
QA5	Is the dataset used clearly identified and mentioned?
QA6	Are the dataset’s characteristic language pairs and its level of granularity (e.g., sentence, paragraph, document) adequately described?
QA7	Does the study describe the full process used for CLPD?
QA8	Is the feature extraction phase clearly defined?
QA9	Does the study present a clear and well-structured experiment?
QA10	Are the obtained results properly interpreted and discussed?

F. Systematic Literature Review (SLR)

SLR is conducted to identify and analyze relevant research on CLPD techniques published between 2014 and 2025. An initial pool of 1300 articles is retrieved from major scientific databases, including IEEE Xplore, ScienceDirect, Springer, Google Scholar, and Scopus. After an initial screening phase, 57 records are excluded for not meeting basic inclusion criteria, specifically, lack of open access, non-English language, or absence of multilingual representation approaches. The remaining 1243 records are then screened more closely, excluding 1160 articles due to duplication or non-relevance based on title and abstract. This process reduces the selected articles for full-text review to 83. At the next stage, 18 articles are excluded for reasons such as lack of experimental validation, being review papers, or omitting baseline or performance metrics. The final 61 articles are assessed in depth for methodological quality, relevance to the research objective, and overall contribution. Ultimately, 43 studies meet all criteria and are included in the final analysis. The entire selection process and applied filters are illustrated in Fig. 1.

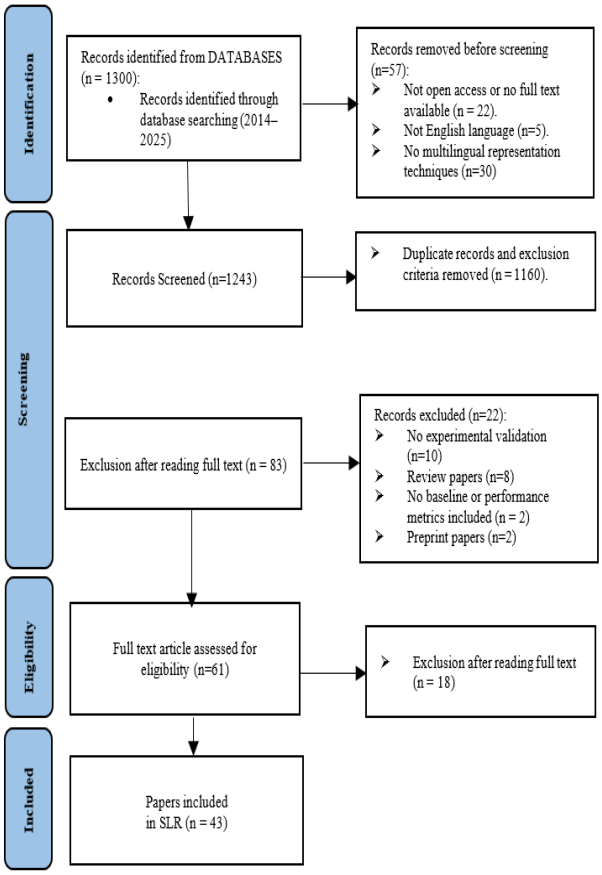


Fig. 1. PRISMA flowchart of the research process.

G. Data Analysis

A total of 43 studies related to the field of CLPD published between 2014 and 2025 are identified. Fig. 2 illustrates the distribution of study types over time. Journal articles dominate the landscape and comprise approximately 57% of the selected studies, followed by conference papers, which represent around 36%, while book chapters constitute the smallest portion, with only 7%. Table III presents the distribution of the retrieved articles across different publication venues. The findings indicate that 59.52% of the studies were published in journals, while 35.71% appeared in conference proceedings. A smaller proportion, 4.76%, was published as book chapters.

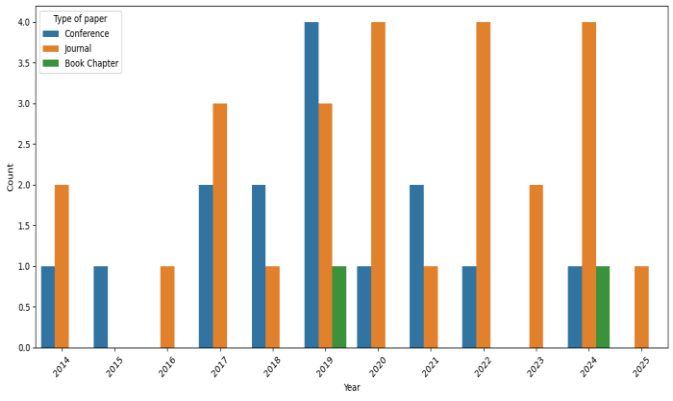


Fig. 2. Distribution of research by type and by year.

TABLE III. DETAILS THE CORRESPONDING PUBLICATION VENUES

Document Type	Publication Title	Reference
Conference Article	Artificial Intelligence XXXVII	[23]
	Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialogue"	[60]
	Proceedings of the 2nd International Conference on Innovation in Artificial Intelligence	[54]
	International Symposium on Communications and Information Technologies (ISCIT)	[38]
	8th SASTech 2014 – Symposium on Advances in Science & Technology	[37]
	Proceedings of the 2018 2nd International Conference on Cloud and Big Data Computing	[45]
	Ivannikov Memorial Workshop (IVMEM)	[66]
	International Conference on Developments of E-Systems Engineering (DeSE)	[40]
	Intelligent Computing	[58]
	Proceedings of the Fourth Arabic Natural Language Processing Workshop	[30]
	Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)	[28]
	Conference on Neural Information Processing Systems (NeurIPS 2019)	[27]
	Workshop on Extraction and Evaluation of Knowledge Entities from Scientific Documents	[47]
	International Workshop on Semantic Evaluation	[25]
	Conference of the European Chapter of the Association for Computational Linguistics	[29]
Journal Article	Information Processing & Management	[32]
	Applied Mechanics and Materials	[33]
	Journal of Information Science	[34]
	Information Processing & Management	[35]
	Intelligent Data Analysis	[36]
	IAES International Journal of Artificial Intelligence (IJ-AI)	[59], [63]
	International Journal of Interactive Mobile Technologies (iJIM)	[64]
	Language Resources and Evaluation	[61]
	Journal of the Association for Information Science and Technology	[42]
	Information Technology Journal	[56]
	The International Journal of Multiphysics	[41]
	Algorithms	[39]
	Journal of Applied Intelligent System	[55]
	Engineering, Technology & Applied Science Research	[57]
	International Journal of Advanced Computer Science and Applications (IJACSA)	[24]
	Asian Journal of Research in Computer Science	[53]

	Proceedings of the Association for Information Science and Technology	[26]
	Data Intelligence	[62]
	Journal of Heuristics	[65]
	Journal of Computing & Biomedical Informatics	[46]
	Journal of AI and Data Mining	[48]
	Journal of King Saud University - Computer and Information Sciences	[52]
	Expert Systems with Applications	[31]
	The European Journal on Artificial Intelligence	[67]
	Journal of Theoretical and Applied Information Technology (JATIT)	[50]
Book Chapter	Recent Advances in Intuitionistic Fuzzy Logic Systems: Theoretical Aspects and Applications	[51]
	Bridging Between Information Retrieval and Databases: Revised Tutorial Lectures of the PROMISE Winter School	[49]

IV. Methods Analysis and Comparison

This section presents the analysis of the research questions, along with the findings derived from the state-of-the-art review. It is structured into two main parts. The first part provides a general overview, examining all the approaches used for CLPD. It highlights the techniques employed, the models applied, and the various language pairs studied. The second part focuses on a detailed and specific analysis of each language pair. For each pair, we evaluate the performance of the employed approaches, considering the linguistic characteristics and datasets used. This in-depth comparative approach highlights the most effective methods for each language pair.

A. Language Pairs (RQ1)

The most commonly used language pairs reveal interesting trends. The En-Ar combination stands out as the most frequently studied, followed by En-Es, En-De, and En-Ar, highlighting the significance of these languages in academic research. In contrast, pairs like Tibetan-Chinese, French-Arabic, and English-Japanese are significantly less explored, due to either a lack of linguistic resources or limited research interest. The omnipresence of English in almost all combinations reflects its dominant position in scientific publications and advancements in natural language processing (NLP). Languages frequently paired with English, such as Arabic, Chinese, or Spanish, are often chosen for their geopolitical importance, widespread use, or linguistic complexity.

It is also relevant to consider the direction of language pairs, for instance, for English-Arabic and Arabic-English, where the source and target languages may differ. This distinction influences the challenges encountered, particularly in terms of machine translation or algorithm adaptation. Furthermore, pairs involving underrepresented languages, such as Tibetan or Persian, often suffer from a lack of annotated corpora and suitable NLP tools, hindering their detailed study. Finally, this analysis reflects not only the current state of research but also the linguistic biases within the field, emphasizing the urgency to diversify efforts to include more marginalized languages and promote linguistic equity on a global scale. Fig. 3 describes the most frequently used language pairs.

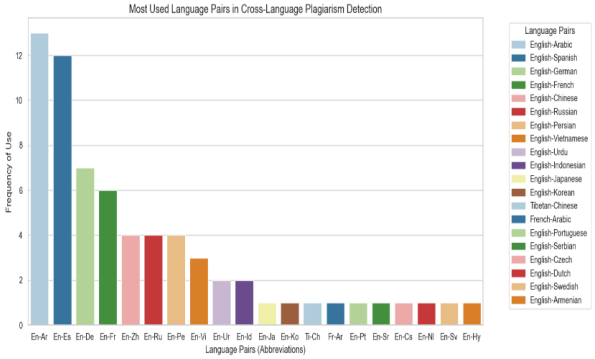


Fig. 3. Frequency of CLPD language Pairs.

B. Translation Approach (RQ2)

In this part, we will examine how translation and non-translation are utilized within the CLPD context, as well as the translation tools employed and the languages being translated.

1) Translation strategies in CLPD: Translation is used in CLPD to convert documents or sentences into the same language, representing a dominant share of 60.5% in the reviewed studies. English is the most studied language, and in most cases, documents are translated into English to ensure both texts are in the same language rather than translated into other languages. Among the translation tools, Google Translate stands out as the most widely utilized due to its accessibility, multilingual support, and continuous improvements in translation quality. However, several studies also explore the use of specialized dictionaries, particularly for domain-specific texts, offering more accurate translations tailored to the subject matter. Additionally, advancements in NLP have led to the adoption of transformer-based models, such as BERT and GPT, which provide context-aware translations and handle complex linguistic structures. Fig. 4 shows the distribution of translation usage.

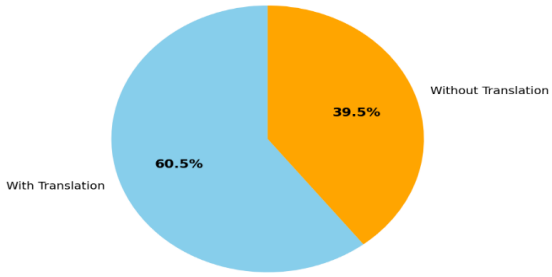


Fig. 4. Distribution of research based on the use of translation.

2) Translation tools used in CLPD: Fig. 5 shows the frequency of translation tools used in CLPD systems, in ascending order of use. Less frequently used tools, such as Transformer, English Expert, BABYLON, and the Dictionary database, appear only once in the studies examined. Tools such as dict. and Bilingual translators are used twice. In contrast, Google Translate stands out as the most widely adopted translation tool, with 21 occurrences, underlining its predominant role and popularity.

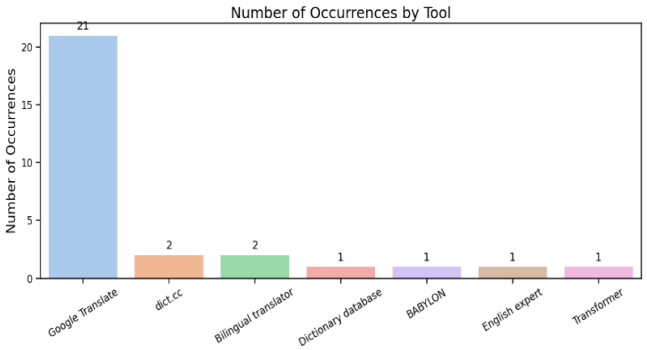


Fig. 5. Frequency of translation tools.

C. Multilingual Text Representation Strategies (RQ3)

This section presents various strategies employed for CLPD, including statistical analysis of approach distribution, a comparison between traditional word embedding methods and transformer-based deep learning models, and the utilization of multilingual semantic networks.

1) Distribution of approaches in CLPD: Fig. 6 highlights the predominance of traditional approaches (47.6%), such as TF-IDF, Word2Vec, LSI, and SVD, which remain widely used due to their simplicity and low computational cost. However, Transformers and Deep Learning models (23.8%) are gaining increasing adoption thanks to their ability to capture rich contextual representations, despite their higher computational requirements. Multilingual Semantic Networks (19%), such as WordNet and BabelNet, play a key role in knowledge-based approaches, particularly for semantic similarity assessment in a multilingual context. Finally, fingerprinting techniques (9.5%), although less common, are employed to capture a unique textual signature.

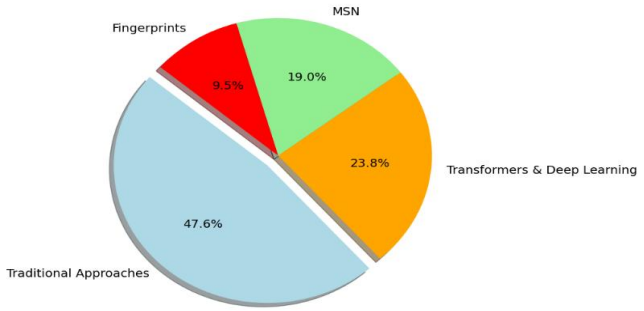


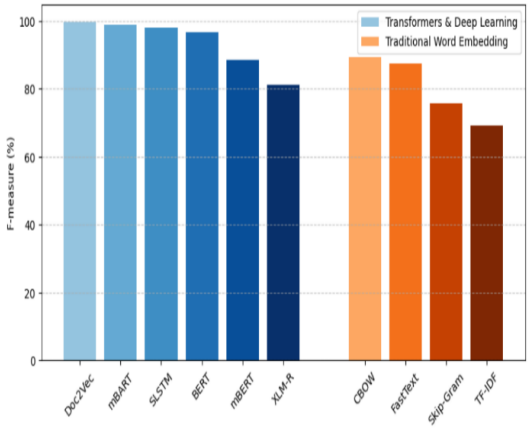
Fig. 6. Distribution of approaches in CLPD.

2) Traditional word embedding methods versus transformers & deep learning: Table IV provides a comparative analysis of traditional word embedding methods (e.g., Word2Vec) and transformer-based deep learning models (e.g., BERT, XLM-R) for CLPD. It includes key details such as the studied language pairs, the algorithms employed, the datasets used for evaluation, and the performance metrics. Although various evaluation metrics such as precision and recall are considered, the F1-score is more commonly used as it provides a balanced measure by incorporating both precision and recall. The F-measure scores, which represent the harmonic mean of precision and recall, are provided to evaluate the effectiveness of these methods. Traditional methods like TF-IDF and Word2Vec often produced results below 80% in terms of F-measure for many language pairs, especially for complex language pairs. For example, TF-IDF/LDA with the En-Es and En-De pairs achieved F-measure scores of 57.58 and 69.27, respectively, while CBOW yielded results of around 78.5. In contrast, Transformer and Deep Learning techniques, such as XLM-R, mBERT, and mBART, surpassed the 80% F-measure mark and demonstrated impressive performance in multilingual contexts, with F-measures ranging from 81 to 98.87, such as 88.5 for En-Vi (XLM-R) and 90.8 for En-De (mBERT). mBART, another transformer model, also performs well, with F-measures of 98.01, 98.87, and 96.02 for the En-Es, En-Fr, and En-De pairs, respectively. In conclusion, transformer-based and deep learning methods generally outperform traditional approaches in terms of F-measure, though results can depend on the specific datasets, and traditional methods remain competitive for simpler tasks or less complex configurations. Fig. 7 compares the performance and temporal evolution of natural language processing models. Fig. 7(a) shows that transformer-based and deep learning models (in blue) generally outperform traditional word embedding methods (in orange) in terms of F-measure, indicating their increased effectiveness. Fig. 7(b) illustrates the timeline of these techniques, revealing that traditional approaches (TF-IDF, Word2Vec) were introduced earlier, while more advanced models like BERT and XLM-R emerged more recently, marking a shift towards more powerful methods over time.

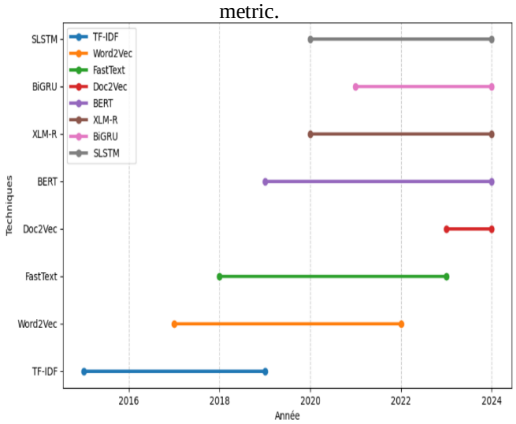
TABLE IV. COMPARISON OF THE PERFORMANCE OF TRADITIONAL WORD EMBEDDING AND TRANSFORMERS & DEEP LEARNING IN CLPD

	ID	Language pair	Algorithm	Dataset	Perf. (%)
Transformers & Deep Learning	[57]	En-Vi	**XLM-R** M-BERT	GLUE	A: 84.3 F1: 88.5
		En-De			A: 87.2 F1: 90.8
		En-Fr			A :86.2 F1 :90.2
	[59]	En-Es	mBART	PAN-PC-11 JRC-ACQUIS EUROPARL Wikipedia Conference papers	A:97.94 P: 98.57 R:97.47 F1:98.01
		En-Fr			A:95.59 P: 95.21 R:96.85 F1 :96.02
		En-De			A: 98.83 P: 98.42 R: 99.32 F1: 98.87
	[60]	En-Ru	BERT	Negative-1	P :96, R :93 F1:95
	[64]	En-Es	Doc2Vec	PAN-PC-11, JRC-ACQUIS EUROPARL, Wikipedia	A:99.81 P:99.75 R:99.88 F1:99.70

	[65]	En-Vi	SLSTM	TED	P :80.3 R :95.8 F1 :87.4
Traditional Word Embedding	[24]	En-Ar	**IDF** CL-WES CBOW	Books, Wikipedia EAPCOUNT MultiUN	**F1:88** F1: 86.5 F1:78.5
	[28]	En-Ar	Word2Vec +TFIDF +MUSE	SemEval- 2017	F1:87.9
	[29]	En-Fr	CBOW	Wikipedia, Conference papers Product reviews, Europarl JRC-Acquis	F1:89.15
	[37]	En-Pe	TF- IDF/VSM	Internet sources	P: 88 R: 96 F1: 91
	[27]	En-Ru	FastText	wikipedia	P: 83 R :79 F1 :80
	[35]	En-Es En-De	TF-IDF	PAN-PC-12, JRC-Acquis PAN-PC-11, Wikipedia	R:78.89 P: 61.74 F1:69.27 F2:74.74
	[32]	En-Es En-De	TF- IDF/LDA	PAN-PC-12	P:46.63, R:75.26 F1:57.58 F2:67.03



(a) Comparison of embedding model performance in terms of the F-measure metric.



(b) Comparison of the embedding model in terms of time evolution.

Fig. 7. Comparison of the performance and temporal evolution of embedding models.

3) Multilingual semantic network: The use of MSN in the field of CLPD serves two main purposes: acting as a dictionary to find synonyms or equivalent concepts across different languages and creating knowledge graphs. According to Fig. 8, we can observe that the majority of research in CLPD does not utilize MSNs, which suggests that many studies rely on alternative approaches, such as techniques based on word embeddings, ML, or DL. However, among the MSNs used, WordNet is the most frequently employed, likely due to its well-established structure and its use as a pivotal lexical resource for several languages. WordNet is often utilized to identify synonyms, antonyms, and other lexical relations, thereby facilitating the detection of semantic similarities. BabelNet, which follows WordNet in usage frequency, stands out for its ability to cover a wide range of languages and integrate information from various sources, such as WordNet and Wikipedia. Other MSNs, such as VietNet, are specifically dedicated to less common languages like Vietnamese, reflecting a focus on local or specific case studies. Finally, Wikidata, a collaborative knowledge base, is less frequently used but offers interesting potential for applications requiring contextualization or encyclopedic information.

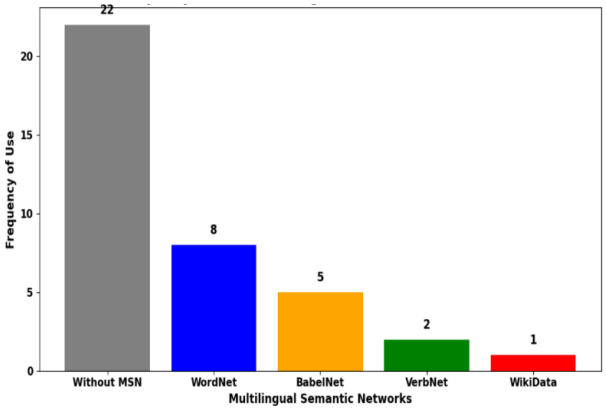


Fig. 8. Frequency of use of multilingual networks.

D. Cross-Lingual Similarity Measures (RQ4)

The most popular similarity metrics in CLPD systems are presented in Fig. 9. Cosine Similarity dominates all studies due to its effectiveness in vector-based representations and its capacity to assess similarities across text lengths. The second most popular index for comparing groups of words or characters is the Jaccard Index. Metrics like WUP similarity, LCS, ED, Lin similarity (Lin), and DC are utilized less often, implying that they are relevant in specific situations or serve as additional methods. Ultimately, the containment measure sees the least use, indicating its specialization in certain scenarios.

E. Comparative Study for each Language Pair (RQ5)

Before analyzing deeply each language pair, the techniques most frequently used in the preprocessing section are discussed, as they are commonly used in the different CLPD approaches.

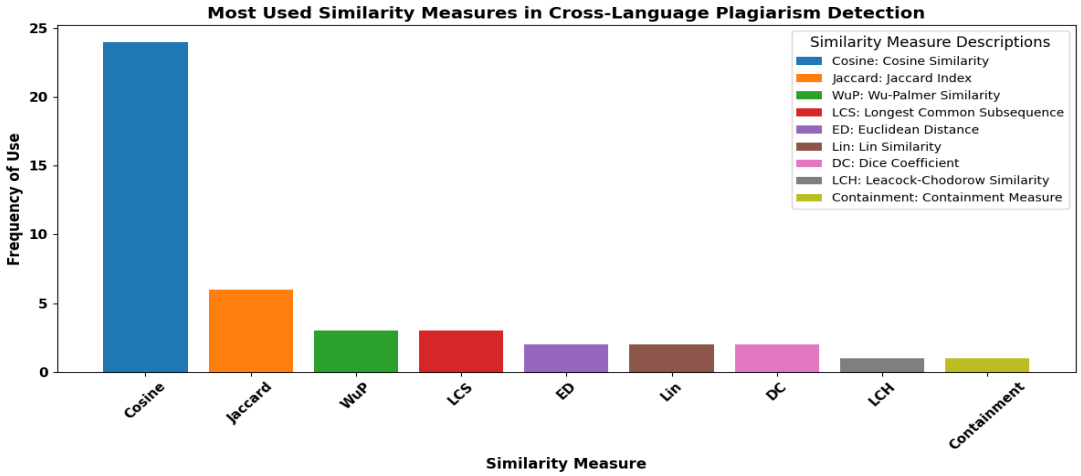


Fig. 9. Frequency of similarity measures in CLPD.

1) Common preprocessing techniques: Text pre-processing in multilingual studies usually begins with the segmentation of the text into sentences or paragraphs, enabling structured and targeted processing. Most of the techniques used at this stage are common and aim to improve data quality. Among the most commonly used steps are tokenization, which divides text into basic units such as words or symbols, and includes converting text to lowercase, removing unnecessary punctuation, lemmatization, and stemming to reduce words to their basic form or root. Stop word removal is another commonly employed technique, where insignificant terms such as articles or prepositions are eliminated to focus on meaningful words. For specific languages, tailored techniques are applied, such as diacritic removal for Arabic or the use of specialized dictionaries and linguistic databases to standardize terms. Additionally, advanced methods like NER are sometimes used to identify key elements such as proper nouns, dates, or locations, enriching data quality for more detailed analysis. These preprocessing steps are common to most studies. Fig. 10, given below, describes the most common preprocessing techniques used.

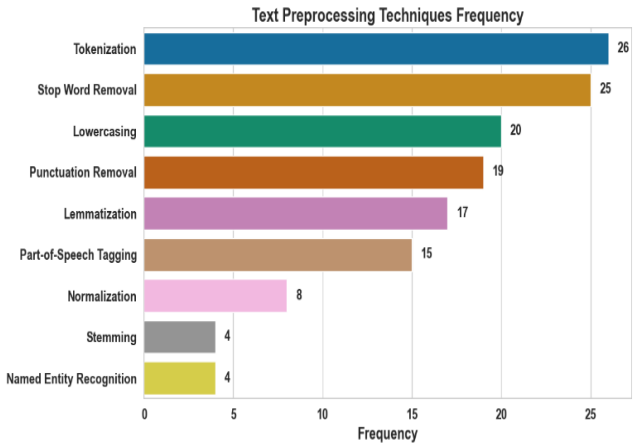


Fig. 10. Most commonly used preprocessing techniques.

2) Analysis of Language Pairs: This section provides an in-depth examination of each language pair, with a particular focus on the datasets employed, the applied techniques, and the performance achieved, while highlighting their unique features and the specific challenges they pose for processing and modeling. The analysis encompasses several key aspects of cross-language plagiarism detection, including the use of translation, the integration of multilingual semantic resources, the methods adopted for feature extraction, and the classification strategies implemented. It also covers the similarity measures applied, the datasets utilized, the levels of granularity considered (ranging from sentence-level to paragraph-level and document-level corpora), and the performance metrics reported. From a global perspective, CLPD research shows a reliance on a set of widely used datasets such as PAN-PC-11, SemEval, JRC-Acquis, Wikipedia, Europarl, and collections of conference papers, with their frequency of use often reflecting the language pairs investigated. In the following sections, we provide a more detailed distribution of datasets for English–Arabic and English–Spanish, which represent the most extensively studied language combinations.

a) English-Arabic language pair: For the En-Ar language pair, the preprocessing methods include tokenization, normalization, removal of stop words (RSW), and diacritic removal for Arabic text. Feature extraction techniques include pre-trained models like Word2Vec, Skip-Gram, CBOW, and TF-IDF. ML models, such as SVM, LR, DT, RF, KNN, and deep neural networks (DNN), are employed, with CS serving as the primary similarity measure. In most studies, Arabic texts are translated into English using automatic translation tools like Google Translate, followed by similarity computations using measures such as CS, ED, or semantic measures like WUP and LCH. For studies utilizing the SemEval-2017 dataset, which focuses on sentences, the combination of translation (Word2Vec+TF-IDF+MUSE) and the SVC model for classification yielded notable results with an F1-score of 87%. PC for the same combination also achieved a score of 81.47%. In datasets like OPUS, KSUCCA, and EAPCOUNT,

integrating BabelNet with a DNN architecture achieved a significant accuracy of 97.01%. Similarly, the use of translation, the fingerprint approach, and the Winnow algorithm with the LLR technique achieved an accuracy of 97%. Some studies also focused solely on measuring similarity percentages between two sentences.

Fig. 11 reveals that the most commonly used datasets for En-Ar language pair research are SemEval-2017 and EAPCOUNT, highlighting their importance for evaluating semantic similarity models and multilingual plagiarism detection. SemEval-2017 is sentence-based, making it suitable for fine-grained similarity tasks, while Wikipedia, being document-based, is more relevant for applications requiring diverse and larger contexts. Additionally, datasets such as articles, tweets, and academic works demonstrate their value in approaches that require varied and realistic text sources. OPUS and KSUCCA, though moderately used, remain significant for aligned multilingual corpora, whereas books and MultiUN complement the range by addressing specific needs. This distribution reflects a preference for datasets that balance accessibility, diversity, and specificity.

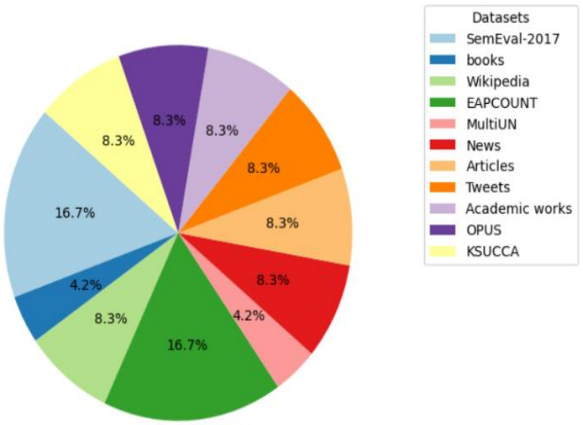


Fig. 11. Most used datasets for the English-Arabic language pair.

b) English-Spanish language pair: For the En-Es language pair, various methodologies, datasets, and performance results are highlighted. The most important approaches leverage advanced transformer models and deep learning architectures. For example, the use of mBART for feature extraction and SLSTM for classification achieved an accuracy of 97.94%, precision of 98.57%, recall of 97.47%, and an F1-score of 98.01%. This method is evaluated using datasets such as Europarl, PAN-PC-11, Wikipedia, JRC-Acquis, and Conference papers. Another notable method used several embedding techniques, including Doc2Vec, GloVe, FastText, BERT, Word2Vec, and Sen2Vec, combined with SLSTM. This strategy achieves accuracy rates between 98.41% and 99.81% on datasets such as JRC-Acquis, PAN-PC-11, Europarl, and Wikipedia. In contrast, methods using traditional feature extraction techniques like TF-IDF and Word2Vec also showed impressive performance. These methods realized plagiarism detection rates of 83.5% and 84.2% on PAN-PC-11 and PAN-PC-12, respectively, and a precision score of 44.3% on SemEval-2017, indicating their efficiency in simpler scenarios. However, approaches centered on knowledge graphs, such as those using Wikidata or BabelNet, displayed moderate results. For example, the method based on Wikidata achieved a plagiarism detection rate of 57.3%, with a precision of 72.3% and a recall of 47.4%. In comparison, the BabelNet-based method recorded a plagiarism detection rate of 60.87%, precision of 70.36%, and recall of 53.99%, both analyzed on the PAN-PC-11 dataset. Similarly, another approach that employed BiGRU and GNN in combination with WordNet and the CS measure achieved a plagiarism detection rate of 62%, precision of 20.3%, and recall of 8.5%.

Fig. 12 illustrates the distribution of datasets used for En-Es. The PAN-PC-11 dataset accounts for the largest proportion at 25%, indicating its extensive use in research. Other datasets, including PAN-PC-12, PAN-PC-14, SemEval-2016, and SemEval-2017, each represent 12.5%, reflecting their consistent contribution to CLPD tasks. Notably, Europarl is used at a smaller rate, 8.3%, while datasets like JRC-Acquis, Wikipedia, MRPC, and conference papers contribute 4.2% each.

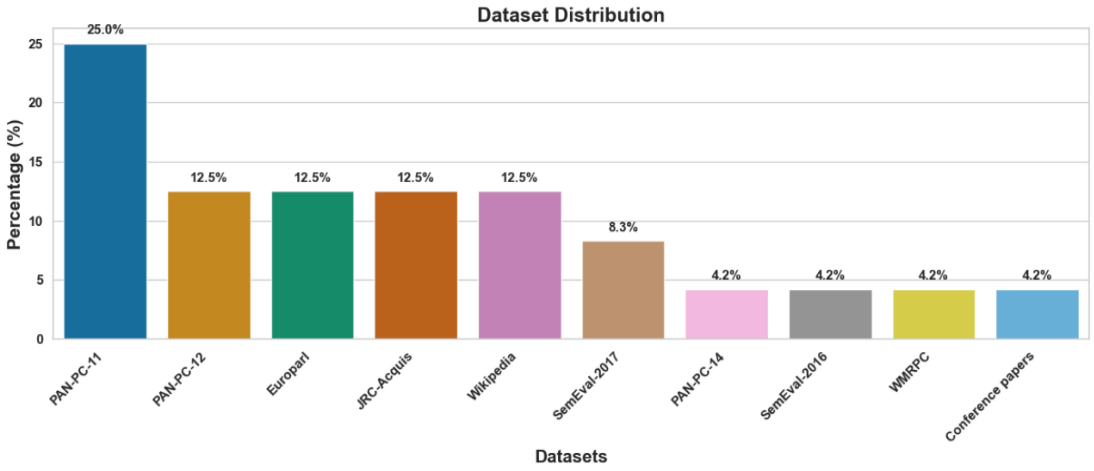


Fig. 12. Most used datasets for the English-Spanish language pair.

c) English-German language pair: For the En-De language pair, the approaches encompass both translation-based and translation-independent methods. The PAN-PC-11 dataset is the most frequently utilized, with the CS measure being widely applied across approaches. Among translation-independent methods, the use of mBART and SLSTM achieves an accuracy of 95.59%, a precision of 95.21%, a recall of 96.85%, and an F1-score of 96.02% across datasets such as PAN-PC-11, JRC-Acquis, Europarl, Wikipedia, and conference papers. For translation-based methods, XLM-R and M-BERT deliver notable results, achieving an accuracy of 87.2% and an F1-score of 90.8% on the GLUE dataset. Conversely, approaches leveraging knowledge graphs, such as Wikidata and BabelNet, demonstrate moderate performance. For instance, the Wikidata-based method achieves a plagiarism detection rate of 52.1%, precision of 67.2%, and recall of 42.5%, while the BabelNet-based approach records a plagiarism detection rate of 52.96%, precision of 63.06%, and recall of 46.71% on the PAN-PC-11 dataset. Additionally, translation-independent methods using traditional feature extraction techniques, including TF-IDF and Word2Vec, show strong performance, with plagiarism detection rates of 85.7% and 86.2% on the PAN-PC-11 and PAN-PC-12 datasets, respectively. In conclusion, transformer models, particularly mBART and SLSTM, demonstrate the most significant results for the En-De language pair, substantially outperforming traditional and knowledge-graph-based approaches.

d) English-French language pair: For the En-Fr language pair, the best-performing approach, translation-independent, combines the mBART transformer for feature extraction with the SLSTM model, using the CS measure. This method achieves exceptional results on datasets such as PAN-PC-11, JRC-Acquis, Europarl, Wikipedia, and conference papers, with an accuracy of 98.83%, a precision of 98.42%, a recall of 99.32%, and an F1-score of 98.87%. Another notable translation-independent approach utilizes CBOW for feature extraction, DT as classifiers, and the CS measure. Tested on datasets including Wikipedia, conference papers, product reviews, Europarl, and JRC-Acquis, this method achieves a solid F1-score of 89.15%. Additionally, a translation-independent approach based on the creation of a knowledge graph using WordNet, combined with BiGRU and GNN models and the CS measure, shows moderate performance. It achieves a precision of 50.6%, a recall of 69%, and a plagiarism detection rate of 58.4% when tested on datasets such as Europarl, JRC-Acquis, and Wikipedia. Another approach, using XLM-R and M-BERT on the GLUE dataset, attains an accuracy of 86.2% and an F1-score of 90.2%.

e) English-Vietnamese language pair: For the English-Vietnamese language pair, both translation-based and translation-independent strategies are emphasized. Translation-based models, such as XLM-R and M-BERT, utilize multilingual embeddings to achieve notable results on the GLUE dataset, with an accuracy of 84.3% and an F1 score of 88.5%. In contrast, translation-free methods leveraging Word2Vec embeddings and lexical resources like WordNet and VietNet demonstrate strong performance, with the SLSTM model achieving an accuracy of 89.61% on the TED dataset. Meanwhile, without word embedding techniques, the use of BabelNet and VietNet with deep learning models (SLSTM) achieves impressive results on the TED dataset, with a precision of 80.3%, a recall of 95.8%, and an F1 score of 87.4%. These findings underscore the effectiveness of both approaches, with translation-free methods often delivering comparable or superior performance.

f) English-Chinese language pair: For the English-Chinese language pair, the approach begins with the use of the Skip-Gram model of Word2Vec and ED as the similarity measure, which attained a high H-score of 97.09% on the NDLTD dataset without relying on translation. Meanwhile, translation-independent methods employing fingerprints and WordNet with the DC deliver a solid performance, achieving a precision of 87% and a recall of 78% on the CNKI dataset. On the other hand, translation-based methods, such as the combination of Google Translate and RCNN with Word2Vec, demonstrate an accuracy of 88.87% on the student dataset. Additionally, the use of Google Translate and BABYLON alongside TF-IDF yielded a Mean Average Precision (MAP) score of 500 on the Xinhua English news dataset.

g) English-Russian language pair: For the English-Russian language pair, all the approaches rely on transformer-based models for the feature extraction phase, achieving results above 80%. The approach using BERT achieves a precision of 96%, outperforming those based on LASER, SE, WS, NMT+1-Gram, and NMT+2-Gram. Furthermore, the use of FastText with CS on the Wikipedia dataset achieves a precision of 83%, a recall of 79%, and an F-score of 80%. Likewise, no use of MSN.

h) English-Persian language pair: For the EN-PE language pair, various translation tools and techniques are implemented, including English experts, Google Translate, and bilingual translators. One notable approach uses the Mizan dataset with WuP as a similarity measure, achieving an accuracy of 98.82%. Google Translate combined with BILBOWA and other techniques like T+MA, CL-LSI, and CL-ESA, leveraging BabelNet for feature extraction, delivers varying precision rates, ranging from 21% to 83% on the HAMTA-CL dataset. The bilingual translator approach employs VSM and TF-IDF with CS, achieving a precision of 88%, a recall of 96%, and an F-score of 91% on Internet sources. Additionally, transformer-based methods like XLM-R, M-BERT, and DistilBERT show interesting precision and similarity scores, with XLM-R achieving 95.62% and 95.17%, M-BERT achieving 91.88% and 91.55%, and DistilBERT achieving 89.51% and 89.08% on the PESTS dataset.

i) English-Urdu language pair: For the English-Urdu language pair, the first approach, based on Google Translate, WordNet, and ML models (KNN, SVM, DT, RF, NB), applied to CLPD-UE-19 with a document-level granularity, achieves an accuracy of 92% with KNN. In contrast, the n-grams + T+MA approach, tested on CLEU (sentence/document level), shows lower performance, with an F1-score of 73.2% for binary classification and 55.2% for ternary classification.

j) English-Indonesian language pair: For the English-Indonesian language pair, two translation-based methods are presented. The first method utilizes Google Translate combined with fingerprinting techniques and the Winnow algorithm, achieving an accuracy of 84.7% on a dataset of scientific articles using the Jaccard similarity measure. The second method relies on a dictionary database, combined with advanced techniques such as LSA, LVQ, and SVD, along with WordNet and the CS measure. This second approach outperforms the first, achieving an accuracy of 87% on the same dataset.

k) Underexplored language Pairs: Certain languages receive limited attention in existing research, primarily because they are not widely spoken or used in international communication. The analysis highlights the underexplored language pairs in CLPD systems, showcasing efforts to address these pairs despite limited resources and research focus. English-Japanese (EN-JA) and English-Korean (EN-KO) achieved promising accuracies of 87.49% and 87.26%, respectively, though with limited datasets. For Traditional Chinese (Ti-Ch), translation-independent methods (Doc2Vec, SLSTM) showed lower precision (54-55%) on datasets like CWMT and SemEval2014. French-Arabic (Fr-Ar), using WordNet similarity measures on datasets of news articles and academic works, recorded a WUP score of 63%. These results reflect the challenges and opportunities for advancing CLPD for less-resourced language pairs.

V. Discussion

This section presents the approaches employed in CLPD, provides a summary of our SLR, and highlights promising research directions that remain largely unexplored.

A. General Architecture for CLPD

In this SLR, we examine the approaches used for CLPD, categorized into four main groups. Similarity-based approaches (MSN) utilize lexical-semantic resources such as WordNet and BabelNet to enhance semantic comparison and measure the relationship between words across languages. These methods also rely on similarity metrics such as cosine similarity, Jaccard index, and Wu-Palmer similarity to assess textual resemblance. Traditional models like TF-IDF and Word2Vec offer a statistical representation of texts. Fingerprinting techniques create distinctive signatures to detect similarities. Lastly, Transformer and deep learning frameworks, including BERT, XLM-R, and mBART, utilize advanced neural architectures for more efficient plagiarism detection.

Fig. 13 illustrates the steps of CLPD systems, which are based on two main strategies: with translation and without translation. In the first approach, the text is translated into a common language to facilitate direct comparison, while in the second approach, detection is performed using multilingual representations without translation. After this initial stage, the text is represented using three methods: Word Embedding (WE alone or WE + MSN), where models like Word2Vec, BERT, or XLM-R transform the text into vectors, either autonomously or by integrating the Multilingual Semantic Network (MSN) to enhance multilingual understanding; Fingerprinting (FP alone or FP + MSN), which generates unique textual fingerprints that can be used alone or combined with MSN to improve semantic alignment; and MSN alone, used independently to detect similarity between texts. After establishing the vector representation of the text, two outputs are obtained: the first output gives a similarity score of two texts via a similarity measure, while the second is a classification using ML or DL algorithms, employing models such as SVM, KNN, or neural networks to distinguish plagiarized from non-plagiarized texts. Finally, a model evaluation phase is carried out using performance measures such as accuracy, precision, recall, and F1 score to assess the effectiveness of the proposed model.

B. Results and Future Research Avenues

This part outlines the key results of the SLR for each inquiry, emphasizing the conclusions derived from the examined studies. It also suggests potential avenues for future investigation, intend to address the identified gaps and steer further advancements in the area.

RQ1: Based on the analysis of the literature review, we have identified 20 language pairs used in work on CLPD. The most frequently studied pairs are En-Ar, En-Es, and En-De, respectively, as shown in Fig. 3. Although other combinations were mentioned, some, such as the English-Hindi pair, were excluded from our selection because the corresponding studies did not meet the inclusion criteria (non-open-source research, review, or monolingual plagiarism). Thus, the total number of pairs actually exploitable remains limited to ten. English appears to be the dominant pivot language, being present in 98% of the language combinations identified.

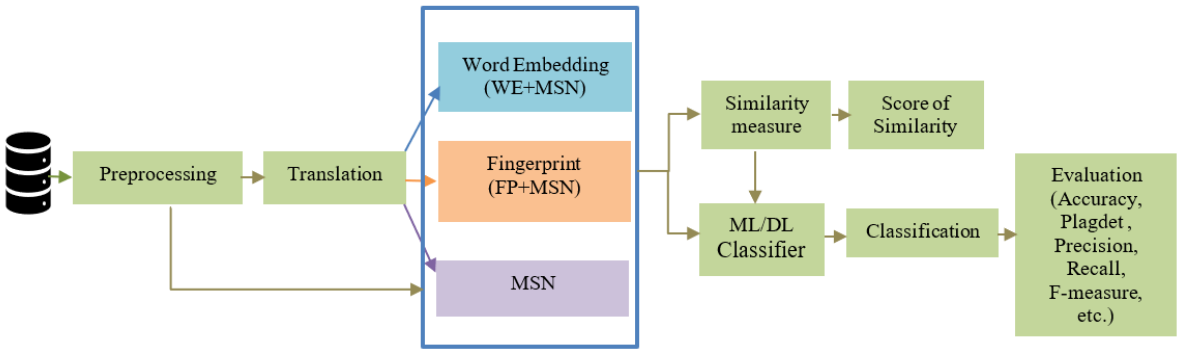


Fig. 13. General architecture for CLPD systems.

RQ2: Concerning the use of translation for CLPD, we observe that during the timeframe of 2014-2025, most documents are translated into the same language, with English being the predominant language for these translations. Additionally, there is a notable portion of studies that are directly incorporated without the process of translation. Among the most widely used translation tools, Google Translate dominates, as illustrated in Fig. 4 and Fig. 5. We also note that for the least explored language pairs, translation is the most used, compared to the most tiled language pair in the context of the CLPD system. According to our analysis, studies that have adopted embedding techniques without going through translation also give significant results, particularly those that have used transformers for different language pairs. The mBART model produced interesting results exceeding 95% for the three language pairs En-Es, En-Fr, and En-De, while the Bert model achieved 96% accuracy for En-Ru. In addition, studies using translation tools achieved interesting results for the English-Persian language pair, with the use of an English expert for translation achieving 98.82% accuracy. In general, the use of advanced embedding techniques, such as transformers, yields interesting results in many languages without the need for translation.

RQ3: Through our review of the literature, we have pinpointed four primary categories of multilingual representation techniques employed in CLPD systems: 1) conventional methods (such as TF-IDF, Word2Vec, FastText), 2) transformer and deep learning architectures (such as mBERT, mBART, XLM-R), 3) multilingual semantic networks like WordNet, BabelNet, VietNet, and Wikidata, and 4) fingerprinting. Historically, conventional methods have been the most prevalent, whereas fingerprinting represents the least investigated area, as depicted in Fig. 6. An analysis of trends over time (Fig. 7) indicates a remarkable transition towards Transformer-based models beginning in 2018, leading to a decline in the use of traditional approaches due to their superior performance and capacity to manage intricate multilingual scenarios. Transformer-based models significantly exceed the performance of traditional word embedding methods, which are often constrained in their F1-measure, as illustrated in Table IV. Furthermore, these models have been evaluated using an expanding range of language pairs, highlighting their versatility. Regarding multilingual semantic networks, WordNet is the most commonly utilized, followed closely by BabelNet, VietNet, and Wikidata (Fig. 8). Nevertheless, these methodologies remain underutilized in CLPD. Lastly, fingerprints are the least frequently employed.

RQ4: Analysis of existing work reveals a diversity of similarity metrics used in CLPD. Nevertheless, the majority of studies favor cosine similarity, due to its effectiveness with vector representations and its ability to evaluate similarities regardless of text length. The second most widely used metric is the Jaccard index, mainly applied to the comparison of sets of words or characters. Other measures, such as Wu-Palmer (WuP), LCS (Longest Common Subsequence), ED (Euclidean Distance), Lin similarity, or the Dice coefficient, appear more marginally, often in specific contexts or as complements to the main methods. Finally, the Containment measure is the least represented, as shown in Fig. 9.

RQ5: In our study, we analyzed several features for each language pair used in CLPD, namely: translation strategy, MSN, feature extraction techniques, methods used (ML/DL), similarity measures, datasets exploited, data granularity, as well as performance obtained. Existing work in CLPD generally pursues two main objectives: similarity score calculation, aimed at comparing sentences or documents using different metrics to evaluate feature extraction techniques, and automatic classification, which consists of training machine or deep learning models to detect cases of plagiarism, relying on metrics such as precision, recall, or F1-measure. Several factors significantly influence the performance of these approaches. Firstly, the granularity of the data plays a decisive role: an analysis at the document level is often more effective than a global analysis of the sentence. Secondly, two main strategies are adopted for multilingual processing: translation into a pivot language, usually English, prior to vectorization, or direct use of the original texts via multilingual models such as mBERT or XLM-R. The choice of method (multilingual semantic networks, fingerprints, traditional embeddings, or Transformers-type models) also has a major impact on the quality of results. Furthermore, poorly endowed languages pose a real challenge due to the lack of suitable linguistic resources, which limits model performance. Finally, advanced models such as BERT, SLSTM, mBART, or XLM-R stand out for their ability to learn rich contextual representations, offering better performance than traditional methods.

Nonetheless, CLPD approaches are still facing several unresolved challenges that manifest across various dimensions of current research:

- Construct diversified multilingual datasets, especially for under-represented languages, and study new and less explored language pairs.
- Analyze and compare the outcomes of methods that incorporate translation against those that do not, while assessing the reliability and performance of various machine translation tools.
- Assess model robustness to paraphrasing, focusing on their capacity to recognize semantically equivalent expressions despite significant syntactic or lexical alterations.
- Examine multiple MSNs such as WordNet, BabelNet, and Wikidata, comparing their performance in terms of semantic richness, contextual precision, and multilingual support.
- Investigate the integration of multiple MSNs (e.g., combining WordNet with BabelNet or Wikidata) to enhance lexical coverage, especially in the context of low-resource languages or domain-specific terminology.
- Evaluate other similarity measures, including LCS, Wu-Palmer, Euclidean distance, Lin, Dice, and cosine similarity, using various types of textual representations.

- Analyze the influence of textual granularity (sentence-level, paragraph-level, or document-level) on detection accuracy and consistency.
- Enhance the capabilities of multilingual knowledge graph-based approaches, focusing on improving their adaptability, scalability, and semantic precision across languages.
- Expand the use of multilingual transformers, given their underutilization in current CLPD studies, to improve scalability and detection accuracy.

VI. Conclusion

Detecting multilingual plagiarism is a major challenge, especially in the academic field. This issue needs a deep study to identify the approaches used for the CLPD system and to respond to many questions. In this context, our work presents an SLR for multilingual plagiarism detection covering the period from 2014 to 2025. Our study proposes four types of multilingual text representations: traditional approaches, multilingual semantic networks, fingerprinting methods, and deep learning models. This review highlights several key findings. The English-Arabic language pair emerges as the most frequently studied, and English appears in 98% of the examined language pairs. Over 60% of the studies incorporate a translation phase, with Google Translate being the most commonly used tool. The mBART model has shown promising results, achieving over 95% accuracy for the En-Es, En-Fr, and En-De language pairs, while the BERT model reached 96% accuracy for the En–Ru pair. Additionally, studies involving translation tools report strong performance for the En-Pe language pair, with an English expert tool for translation reaching up to 98.82% accuracy. Overall, the use of advanced embedding techniques such as transformers has yielded strong results across various language pairs, often without requiring translation. However, it is important to note that the best results do not depend only on the approach used. Firstly, the granularity of the dataset plays a crucial role: detecting plagiarism in a document is different from detecting plagiarism in a sentence. Secondly, the choice of language pair has a significant impact: little-studied languages generally achieve poorer results due to a lack of linguistic resources and research attention. Thirdly, the type of the adopted approach influences the performance. All these factors have a significant impact on CLPD system performance. For future work, we aim to develop universal CLPD models capable of handling any language and form of plagiarism. A promising direction is to leverage knowledge graph–based text representations, enabling the detection of subtle paraphrasing and improving cross-lingual generalization.

References

[1] T. Foltýnek, N. Meuschke, et B. Gipp, “Academic Plagiarism Detection: A Systematic Literature Review”, ACM Comput. Surv., vol. 52, no 6, p. 1-42, nov. 2020, doi: 10.1145/3345317.

[2] O. Zimba et A. Y. Gasparyan, “Plagiarism detection and prevention: a primer for researchers”, doi: 10.5114/reum.2021.105974.

[3] H. E. Mostafa et F. Benabbou, “A deep learning based technique for plagiarism detection: a comparative study”, IAES Int. J. Artif. Intell. IJ-AI, vol. 9, no 1, Art. no 1, mars 2020, doi: 10.11591/ijai.v9.i1.pp81-90.

[4] S. Sierra-Martínez, M.-E. Martínez-Figueira, M. D. Castro Pais, et T. Pessoa, “You work, I copy’. Images, narratives and metaphors around academic plagiarism through Fotovoz”, Br. Educ. Res. J., vol. 50, no 3, p. 1514-1532, 2024, doi: 10.1002/berj.3977.

[5] S. P. J. M. (Serge) Horbach et W. (Willem) Halffman, “The extent and causes of academic text recycling or ‘self-plagiarism’”, Res. Policy, vol. 48, no 2, p. 492-502, mars 2019, doi: 10.1016/j.respol.2017.09.004.

[6] F. Alvi, M. Stevenson, et P. Clough, “Paraphrase type identification for plagiarism detection using contexts and word embeddings”, Int. J. Educ. Technol. High. Educ., vol. 18, no 1, p. 42, août 2021, doi: 10.1186/s41239-021-00277-8.

[7] K. Avetisyan, G. Gritsay, et A. Grabovoy, “Cross-Lingual Plagiarism Detection: Two Are Better Than One”, Program. Comput. Softw., vol. 49, no 4, p. 346-354, août 2023, doi: 10.1134/S0361768823040138.

[8] O. Bakhteev et al., “Cross-Language Plagiarism Detection: A Case Study of European Languages Academic Works”, in Academic Integrity: Broadening Practices, Technologies, and the Role of Students: Proceedings from the European Conference on Academic Integrity and Plagiarism 2021, S. Bjelobaba, T. Foltýnek, I. Glendinning, V. Krásničan, et D. H. Dlabolová, Éd., Cham: Springer International Publishing, 2022, p. 143-161. doi: 10.1007/978-3-031-16976-2_9.

[9] M. AlSallal, R. Iqbal, S. Amin, A. James, et V. Palade, “An Integrated Machine Learning Approach for Extrinsic Plagiarism Detection”, in 2016 9th International Conference on Developments in eSystems Engineering (DeSE), août 2016, p. 203-208. doi: 10.1109/DeSE.2016.1.

[10] M. Singh et V. Gupta, “Review of Extrinsic Plagiarism Detection Techniques and Their Efficiency Comparison”, in Advanced Network Technologies and Intelligent Computing, I. Woungang, S. K. Dhurandher, K. K. Pattanaik, A. Verma, et P. Verma, Éd., Cham: Springer International Publishing, 2022, p. 609-624. doi: 10.1007/978-3-030-96040-7_46.

[11] M. AlSallal, R. Iqbal, V. Palade, S. Amin, et V. Chang, “An integrated approach for intrinsic plagiarism detection”, Future Gener. Comput. Syst., vol. 96, p. 700-712, juill. 2019, doi: 10.1016/j.future.2017.11.023.

[12] M. F. Manzoor, M. S. Farooq, M. Haseeb, U. Farooq, S. Khalid, et A. Abid, “Exploring the Landscape of Intrinsic Plagiarism Detection: Benchmarks, Techniques, Evolution, and Challenges”, IEEE Access, vol. 11, p. 140519-140545, 2023, doi: 10.1109/ACCESS.2023.3338855.

[13] R. Pandit, S. Sengupta, S. K. Naskar, N. S. Dash, et M. M. Sardar, “Improving Semantic Similarity with Cross-Lingual Resources: A Study in Bangla—A Low Resourced Language”, Informatics, vol. 6, no 2, Art. no 2, juin 2019, doi: 10.3390/informatics6020019.

[14] D. G. Dusza, “Machine Translation in the Writing Process: Pedagogy, Plagiarism, Policy, and Procedures”, in Second Handbook of Academic Integrity, S. E. Eaton, Éd., Cham: Springer Nature Switzerland, 2024, p. 1487-1509. doi: 10.1007/978-3-031-54144-5_152.

[15] V. Vysotska, Y. Burov, V. Lytvyn, et A. Demchuk, “Defining Author’s Style for Plagiarism Detection in Academic Environment”, in 2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP), août 2018, p. 128-133. doi: 10.1109/DSMP.2018.8478574.

[16] J. P. Wahle, T. Ruas, T. Foltýnek, N. Meuschke, et B. Gipp, “Identifying Machine-Paraphrased Plagiarism”, in Information for a Better World: Shaping the Global Future, M. Smits, Éd., Cham: Springer International Publishing, 2022, p. 393-413. doi: 10.1007/978-3-030-96957-8_34.

[17] H. Ezzikouri, M. Erritali, et M. Oukessou, “Plagiarism Detection in Across Less Related Languages (English-Arabic): A Comparative Study”, in Smart Data and Computational Intelligence, F. Khoukhi, M. Bahaj, et M. Ezziyyani, Éd., Cham: Springer International Publishing, 2019, p. 207-213. doi: 10.1007/978-3-030-11914-0_22.

[18] F. M. Prentice et C. E. Kinden, “Paraphrasing tools, language translation tools and plagiarism: an exploratory study”, Int. J. Educ. Integr., vol. 14, no 1, p. 11, déc. 2018, doi: 10.1007/s40979-018-0036-7.

[19] Nirbhay Kumar Chaubey Et Al, “Automatic plagiarism detection and extraction in a multilingual: a critical study and comparison”, janv. 2022, doi: 10.17605/OSF.IO/DWUK4.

[20] M. U. Akhtar, J. Liu, Z. Xie, X. Liu, S. Ahmed, et B. Huang, “Entity alignment based on relational semantics augmentation for multilingual knowledge graphs”, Knowl.-Based Syst., vol. 252, p. 109494, sept. 2022, doi: 10.1016/j.knosys.2022.109494.

[21] M. Elkhidir, M. M. Ibrahim, T. A. Khalid, S. Ibrahim, et M. Awadalla, “Plagiarism detection using free-text fingerprint analysis”, in 2015 World

Symposium on Computer Networks and Information Security (WSCNIS), sept. 2015, p. 1-4. doi: 10.1109/WSCNIS.2015.7368306.

[22] M. A. Botto Tobar, M. G. J. van den Brand, et A. Serebrenik, “Cross-Language Plagiarism Detection: Methods, Tools, and Challenges: A Systematic Review”, Int. J. Adv. Sci. Eng. Inf. Technol., vol. 12, no 2, p. 589-599, mai 2022, doi: 10.18517/ijaseit.12.2.14711.

[23] N. Alotaibi et M. Joy, “Using Sentence Embedding for Cross-Language Plagiarism Detection”, in Artificial Intelligence XXXVII, M. Bramer et R. Ellis, Éd., Cham: Springer International Publishing, 2020, p. 373-379. doi: 10.1007/978-3-030-63799-6_28.

[24] H. Aljuaid, “Cross-Language Plagiarism Detection using Word Embedding and Inverse Document Frequency (IDF)”, Int. J. Adv. Comput. Sci. Appl. IJACSA, vol. 11, no 2, Art. no 2, 32/29 2020, doi: 10.14569/IJACSA.2020.0110231.

[25] J. Ferrero, F. Agnes, L. Besacier, et D. Schwab, “CompiLIG at SemEval-2017 Task 1: Cross-Language Plagiarism Detection Methods for Semantic Textual Similarity”, 5 avril 2017, arXiv: arXiv:1704.01346. doi: 10.48550/arXiv.1704.01346.

[26] C.-M. Chang, C.-H. Chang, et S.-Y. Hwang, “Employing word mover’s distance for cross-lingual plagiarized text detection”, Proc. Assoc. Inf. Sci. Technol., vol. 57, no 1, p. e229, 2020, doi: 10.1002/pra2.229.

[27] R. Kuznetsova, “CrossLang: the system of cross-lingual plagiarism detection”, janv. 2019.

[28] N. Alotaibi et M. Joy, “English-Arabic cross-language plagiarism detection”, in Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021), Online: INCOMA Ltd, sept. 2021, p. 44-52. doi: 10.26615/978-954-452-072-4_006.

[29] J. Ferrero, F. Agnes, L. Besacier, et D. Schwab, “UsingWord Embedding for Cross-Language Plagiarism Detection”, 10 février 2017, arXiv: arXiv:1702.03082. doi: 10.48550/arXiv.1702.03082.

[30] R. Lachraf, E. M. B. Nagoudi, Y. Ayachi, A. Abdelali, et D. Schwab, “ArbEngVec : Arabic-English Cross-Lingual Word Embedding Model”, in Proceedings of the Fourth Arabic Natural Language Processing Workshop, W. El-Hajj, L. H. Belguith, F. Bougares, W. Magdy, I. Zitouni, N. Tomeh, M. El-Haj, et W. Zaghouani, Éd., Florence, Italy: Association for Computational Linguistics, août 2019, p. 40-48. doi: 10.18653/v1/W19-4605.

[31] M. Roostaee, S. M. Fakhrahmad, et M. H. Sadreddini, “Cross-language text alignment: A proposed two-level matching scheme for plagiarism detection”, Expert Syst. Appl., vol. 160, p. 113718, déc. 2020, doi: 10.1016/j.eswa.2020.113718.

[32] N. Ehsan et A. Shakery, “Candidate document retrieval for cross-lingual plagiarism detection using two-level proximity information”, Inf. Process. Manag., vol. 52, no 6, p. 1004-1017, nov. 2016, doi: 10.1016/j.ipm.2016.04.006.

[33] H. Y. Chen et P. Vines, “Multi Queries Methods of the Chinese-English Bilingual Plagiarism Detection”, Appl. Mech. Mater., vol. 462-463, p. 1158-1162, 2014, doi: 10.4028/www.scientific.net/AMM.462-463.1158.

[34] N. Ehsan, A. Shakery, et F. Tompa, “Cross-lingual text alignment for fine-grained plagiarism detection”, J. Inf. Sci., vol. 45, p. 016555151878769, août 2018, doi: 10.1177/0165551518787696.

[35] M. Roostaee, M. H. Sadreddini, et S. M. Fakhrahmad, “An effective approach to candidate retrieval for cross-language plagiarism detection: A fusion of conceptual and keyword-based schemes”, Inf. Process. Manag., vol. 57, no 2, p. 102150, mars 2020, doi: 10.1016/j.ipm.2019.102150.

[36] H. Asghari, O. Fatemi, S. Mohtaj, H. Faili, et P. Rosso, “On the use of word embedding for cross language plagiarism detection”, Intell. Data Anal., vol. 23, no 3, p. 661-680, janv. 2019, doi: 10.3233/IDA-183985.

[37] S. Enayati, “Introducing an Automated Technique for Bilingual Plagiarism detection of English-Persian Documents”, 2014.

[38] L. T. Nguyen et D. Dien, “Vietnamese- English Cross-Lingual Paraphrase Identification Using Siamese Recurrent Architectures”, in 2019 19th International Symposium on Communications and Information Technologies (ISCIT), sept. 2019, p. 70-75. doi: 10.1109/ISCIT.2019.8905116.

[39] A. A. P. Ratna et al., “Cross-Language Plagiarism Detection System Using Latent Semantic Analysis and Learning Vector Quantization”, Algorithms, vol. 10, p. 69, juin 2017, doi: 10.3390/a10020069.

[40] E. Hattab, “Cross-Language Plagiarism Detection Method: Arabic vs. English”, in 2015 International Conference on Developments of E-Systems Engineering (DeSE), déc. 2015, p. 141-144. doi: 10.1109/DeSE.2015.25.

[41] J. Min, “Cross-Language Translation Algorithm Based on Word Vector and Syntactic Analysis”, Int. J. Multiphysics, vol. 18, no 2, Art. no 2, avr. 2024.

[42] I. Muneer, M. Sharjeel, M. Iqbal, R. M. A. Nawab, et P. Rayson, “CLEU - A Cross-language english-urdu corpus and benchmark for text reuse experiments”, J. Assoc. Inf. Sci. Technol., vol. 70, no 7, p. 729-741, 2019, doi: 10.1002/asi.24074.

[43] M. AlMousa, R. Benlamri, et R. Khoury, “A novel word sense disambiguation approach using WordNet knowledge graph”, Comput. Speech Lang., vol. 74, p. 101337, juill. 2022, doi: 10.1016/j.csl.2021.101337.

[44] R. Navigli et S. P. Ponzetto, “BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network”, Artif. Intell., vol. 193, p. 217-250, déc. 2012, doi: 10.1016/j.artint.2012.07.001.

[45] H. Ezzikouri, M. Oukessou, M. Youness, et M. Erritali, “Fuzzy Cross Language Plagiarism Detection (Arabic-English) using WordNet in a Big Data environment”, in Proceedings of the 2018 2nd International Conference on Cloud and Big Data Computing, in ICCBDC ’18. New York, NY, USA: Association for Computing Machinery, août 2018, p. 22-27. doi: 10.1145/3264560.3264562.

[46] M. M. Zahid, K. Abid, A. Rehman, M. Fuzail, et N. Aslam, “An Efficient Machine Learning Approach for Plagiarism Detection in Text Documents”, J. Comput. Biomed. Inform., vol. 4, no 02, Art. no 02, mars 2023, doi: 10.56979/402/2023.

[47] J. Stegmüller, F. Bauer-Marquart, N. Meuschke, T. Ruas, M. Schubotz, et B. Gipp, “Detecting Cross-Language Plagiarism using Open Knowledge Graphs”, p. 853881 Bytes, 2021, doi: 10.6084/m9.figshare.17212340.v3.

[48] F. Safi-Esfahani, S. Rakian, et M. H. Nadimi-Shahraki, “English-Persian Plagiarism Detection based on a Semantic Approach”, J. AI Data Min., vol. 5, no 2, p. 275-284, juill. 2017, doi: 10.22044/jadm.2016.770.

[49] M. Franco Salvador, P. Gupta, et P. Rosso, “Knowledge Graphs as Context Models: Improving the Detection of Cross-Language Plagiarism with Paraphrasing”, 2014, p. 227-236. doi: 10.1007/978-3-642-54798-0_12.

[50] M. H. A. Almayali, Z. Alaa, et S. Tiun, Cross Language Plagiarism of Arabic-English Documents Using Linear Log. Noor Publishing, 2017.

[51] H. Ezzikouri, M. Oukessou, M. Erritali, et Y. Madani, “Fuzzy Cross Language Plagiarism Detection Approach Based on Semantic Similarity and Hadoop MapReduce”, in Recent Advances in Intuitionistic Fuzzy Logic Systems: Theoretical Aspects and Applications, S. Melliani et O. Castillo, Éd., Cham: Springer International Publishing, 2019, p. 181-190. doi: 10.1007/978-3-030-02155-9_15.

[52] S. Alzahrani et H. Aljuaid, “Identifying cross-lingual plagiarism using rich semantic features and deep neural networks: A study on Arabic-English plagiarism cases”, J. King Saud Univ. - Comput. Inf. Sci., vol. 34, no 4, p. 1110-1123, avr. 2022, doi: 10.1016/j.jksuci.2020.04.009.

[53] M. Al-Suhaiqi, M. A. S. Hazaa, et M. Albared, “Arabic English Cross-Lingual Plagiarism Detection Based on Keyphrases Extraction, Monolingual and Machine Learning Approach”, Asian J. Res. Comput. Sci., vol. 2, no 3, Art. no 3, févr. 2019, doi: 10.9734/ajrcos/2018/v2i330075.

[54] L. Gang, Z. Quan, et L. Guang, “Cross-language plagiarism detection based on WordNet”, in Proceedings of the 2nd International Conference on Innovation in Artificial Intelligence, in ICIAI ’18. New York, NY, USA: Association for Computing Machinery, mars 2018, p. 163-168. doi: 10.1145/3194206.3194222.

[55] M. Mentari, I. Rozi, et M. Rahayu, “Cross-Language Text Document Plagiarism Detection System Using Winnowing Method”, J. Appl. Intell. Syst., vol. 7, p. 44-57, mai 2022, doi: 10.33633/jais.v7i1.5950.

[56] A. Aljohani et M. Mohd, “Arabic-English Cross-language Plagiarism Detection using Winnowing Algorithm”, Inf. Technol. J., vol. 13, p. 2349-2355, déc. 2014, doi: 10.3923/itj.2014.2349.2355.

[57] H. V. T. Chi, D. L. Anh, N. L. Thanh, et D. Dinh, “English-Vietnamese Cross-Lingual Paraphrase Identification Using MT-DNN”, Eng. Technol. Appl. Sci. Res., vol. 11, no 5, Art. no 5, oct. 2021, doi: 10.48084/etasr.4300.

[58] M. Alshehri, N. Beloff, et M. White, “AraXLM: New XLM-RoBERTa Based Method for Plagiarism Detection in Arabic Text”, in Intelligent Computing, K. Arai, Éd., Cham: Springer Nature Switzerland, 2024, p. 81-96. doi: 10.1007/978-3-031-62277-9_6.

[59] C. Bouaine et F. Benabbou, “Efficient cross-lingual plagiarism detection using bidirectional and auto-regressive transformers”, IAES Int. J. Artif. Intell. IJ-AI, vol. 13, no 4, Art. no 4, déc. 2024, doi: 10.11591/ijai.v13.i4.pp4619-4629.

[60] V. ZubarevD et V. SochenkovI, “Cross-language text alignment for plagiarism detection based on contextual and context-free models”, 2019.

[61] M. Abdous, P. Piroozfar, et B. MinaeiBidgoli, “PESTS: Persian_English cross lingual corpus for semantic textual similarity”, Lang. Resour. Eval., août 2024, doi: 10.1007/s10579-024-09759-3.

[62] W. Bao, J. Dong, Y. Xu, Y. Yang, et X. Qi, “Exploring Attentive Siamese LSTM for Low-Resource Text Plagiarism Detection”, Data Intell., vol. 6, no 2, p. 488-503, mai 2024, doi: 10.1162/dint_a_00242.

[63] O. Hourrane et E. H. Benlahmar, “Graph transformer for cross-lingual plagiarism detection”, IAES Int. J. Artif. Intell. IJ-AI, vol. 11, no 3, Art. no 3, sept. 2022, doi: 10.11591/ijai.v11.i3.pp905-915.

[64] C. Bouaine, F. Benabbou, et I. Sadgali, “Word Embedding for High Performance Cross-Language Plagiarism Detection Techniques”, Int. J. Interact. Mob. Technol. IJIM, vol. 17, no 10, Art. no 10, mai 2023, doi: 10.3991/ijim.v17i10.38891.

[65] D. Dinh et N. L. Thành, “English–Vietnamese cross-language paraphrase identification using hybrid feature classes”, J. Heuristics, vol. 28, avr. 2022, doi: 10.1007/s10732-019-09411-2.

[66] T. Ter-Hovhannisyan et K. Avetisyan, “Transformer-Based Multilingual Language Models in Cross-Lingual Plagiarism Detection”, in 2022 Ivannikov Memorial Workshop (IVMEM), sept. 2022, p. 72-80. doi: 10.1109/IVMEM57067.2022.9983968.

[67] I. Muneer, N. Waheed, M. A. Ashraf, et R. M. Adeel Nawab, “Sentential Cross-lingual Paraphrase Detection for English-Urdu Language Pair”, Eur. J. Artif. Intell., p. 30504554251319446, mars 2025, doi: 10.1177/30504554251319446.

[68] J. Memon, M. Sami, R. A. Khan, et M. Uddin, “Handwritten Optical Character Recognition (OCR): A Comprehensive Systematic Literature Review (SLR)”, IEEE Access, vol. 8, p. 142642-142668, 2020, doi: 10.1109/ACCESS.2020.3012542.

[69] L. E. Jiani, S. E. Filali, E. H. B. Lahmar, et I. Haloum, “Deep Learning-Based Approaches Using Medical Imaging for Therapy Response Prediction in Breast Cancer: A Systematic Literature Review”, Int. J. Online Biomed. Eng. IJOE, vol. 20, no 12, Art. no 12, sept. 2024, doi: 10.3991/ijoe.v20i12.49709.

[70] L. E. Jiani, S. E. Filali, E. H. B. Lahmar, et I. Haloum, “Deep Learning-Based Approaches Using Medical Imaging for Therapy Response Prediction in Breast Cancer: A Systematic Literature Review”, Int. J. Online Biomed. Eng. IJOE, vol. 20, no 12, Art. no 12, sept. 2024, doi: 10.3991/ijoe.v20i12.49709.

Towards Explainable and Balanced Federated Learning: A Neural Network Approach for Multi-Client Fraud Detection

Nurafni Damanik[1], Chuan-Ming Liu[2]*
College of Electrical Engineering and Computer Science, National Taipei University of Technology, Taipei, Taiwan[1]
Department of Computer Science and Information Science, National Taipei University of Technology, Taipei, Taiwan[2]

***Abstract*—The growing demand for secure and privacy-preserving machine learning frameworks has resulted in the implementation of federated learning (FL), especially in critical areas like Credit card fraud detection. This study presents a comprehensive federated learning architecture that incorporates Neural Networks as local models, in conjunction with KMeans-SMOTEENN to address class imbalance in distributed datasets. The system utilises the Flower framework, employing the FedAvg algorithm across ten decentralised clients to collectively train the global model while preserving raw data confidentiality. To improve model transparency and cultivate stakeholder trust, Local Interpretable Model-Agnostic Explanations (LIME) is utilized, offering localised, comprehensible insights into model decisions. The experimental results indicate that the suggested method effectively achieves high predictive accuracy and explainability, rendering it appropriate for real-world fraud detection contexts that necessitate data confidentiality and model accountability.**

***Keywords*—Component federated learning; K-Means SMOTEENN; credit card fraud detection; LIME**

I. Introduction

In recent years, the proliferation of global communication and advancements in computing technology have significantly contributed to the widespread use of credit card transactions. However, this growth has been accompanied by a surge in fraudulent credit card activities. According to data reported by the European Central Bank, Europe experiences annual financial losses amounting to billions of Euros as a result of credit cards [1].

Traditional fraud detection methods, which typically rely on rule-based systems or manual monitoring, have become increasingly inadequate due to their limited ability to dynamically adapt to evolving fraudulent patterns and their reliance on centralized data storage, raising significant privacy and regulatory concerns [2].

Consequently, researchers and practitioners have progressively shifted towards machine learning (ML) methods, which offer superior predictive capabilities by analyzing vast datasets and recognizing complex fraud patterns. While these ML-based approaches, including deep learning models such as CNN, LSTM, and Autoencoders, have shown promising results [3], the centralized nature of data processing inherent in these models poses significant challenges, particularly regarding data privacy, security, and regulatory compliance in sensitive sectors such as finance [4].

To address these critical limitations, federated learning (FL) has emerged as an innovative solution, enabling decentralized model training without exposing sensitive raw data. FL allows multiple institutions to collaboratively train global models by aggregating locally trained model parameters, significantly enhancing data privacy and security [5]. Despite its advantages, current FL implementations often face issues such as data imbalance and lack of model interpretability, critical aspects that affect real-world applicability, particularly in fraud detection scenarios [6]. How can we design a privacy-preserving and interpretable federated fraud detection system that remains effective under severe class imbalance and cross-client data heterogeneity, while satisfying regulatory constraints and enabling real-world deployment?

This study aims to: 1) develop a multi-client FL pipeline (FedAvg) that trains a global model without centralizing sensitive data; 2) address class imbalance at the client level by implementing a client-side hybrid resampling technique (KMeans-SMOTEENN) on each client's training subset prior to local training; validation and test datasets remain unaltered, no raw samples are transmitted from the client, and solely updated model parameters are dispatched for FedAvg aggregation; 3) compare six deep learning architectures—FNN, DNN, CNN, LSTM, Autoencoder, and a stacked DL model within the same FL pipeline to identify the best performer for fraud detection; 4) enhance interpretability by applying LIME to the top-performing model (by AUPRC) and analyzing feature attributions to support audit and domain validation; and 5) compare the performance of our proposed method with state-of-the-art approaches.

Regardless of its benefits, current research on FL-based fraud detection encounters two principal challenges: i) significant class imbalance and non-IID data distributions among clients, which obstruct the identification of minority-class fraud cases, and ii) restricted interpretability, which impedes auditing, regulatory compliance, and stakeholder confidence. Moreover, there is scant information concerning the efficacy of hybrid resampling techniques (e.g., KMeans-SMOTEENN) in federated environments, as well as the comparative performance of various deep learning architectures for rare-event fraud detection. These shortcomings present substantial obstacles to the dependable and regulatory-

*Corresponding Author.

compliant use of FL in practical financial settings.

The proposed framework advances regulation-aligned fraud detection by enabling cross-institution collaboration without data sharing, improving minority-class detection under severe imbalance and non-IID partitions, and closing the explainability gap via LIME. The result is a deployable recipe FL with client-side hybrid resampling, neural models, and post-hoc explanations that: i) raises recall/AUPRC for rare fraud events, ii) preserves privacy and supports compliance, iii) provides decision transparency for risk and audit teams, and iv) generalizes across heterogeneous clients. The approach is transferable to other sensitive domains (e.g., finance and healthcare), where privacy, imbalance, and interpretability are critical.

Building on these objectives, this work makes the following contributions:

1) We offer an innovative federated learning framework that integrates the FedAvg algorithm with KMeans-SMOTEENN, effectively tackling significant class imbalance and heterogeneity in distant client datasets.

2) Thorough Evaluation of Deep Learning Models: We rigorously assess and compare the efficacy of six different deep learning models (FNN, DNN, CNN, LSTM, Autoencoder, and Stacked-DL) in our federated learning framework, yielding critical insights for optimal model selection in fraud detection applications.

3) We utilize Local Interpretable Model-agnostic Explanations (LIME) on the top-performing model (highest AUPRC) to improve the interpretability and transparency of its predictions. This focused strategy guarantees that stakeholders comprehend feature contributions in essential projections.

4) Empirical Validation: Through thorough experimentation, we experimentally confirm that our proposed federated learning technique regularly outperforms standard centralized approaches in accuracy, recall, F1-score, AUC, and AUPRC.

This is how the rest of the study is structured: Section II reviews related work on centralized and federated credit card fraud detection, including class-imbalance remedies and model explainability. Section III details the materials and methods: the overall workflow, dataset, six neural architectures (FNN, DNN, CNN, LSTM, Autoencoder, and a stacked model), the KMeans-SMOTEENN resampling strategy, and the federated learning setup based on FedAvg and the system architecture. Section IV presents experimental results and discussion, including global evaluations, LIME-based explanations, and a comparison with state-of-the-art methods. Section V concludes the study and outlines directions for future work.

II. Related Work

Fraud detection algorithms employ machine learning to effectively identify fraudulent transactions. The majority of suggested CCFDS utilize centralized learning models, whereas a few academics are developing federated learning models to address fraud detection. The supervised, unsupervised, and semi-supervised learning models employ centralized learning algorithms [6].

Credit card fraud detection has been addressed through a variety of machine learning and deep learning techniques, encompassing both supervised and unsupervised methods. Khalid et al. [7] utilized a combination of supervised machine learning models, including Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Decision Tree (DT), Random Forest (RF), Bagging, and Boosting. Alrashdi et al. [8] concentrated on supervised machine learning techniques, employing Random Forest (RF), Extreme Gradient Boosting (XGBoost), and Light Gradient Boosting Machine (LightGBM) to identify fraudulent transactions. Feng and Kim [9] employed a combination of machine learning models, including Random Forest with AdaBoost (RF + AB), Gradient Boosted Decision Trees (GBDT), Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Convolutional Neural Network (CNN).

Desai and Hase [3] undertook a review of deep learning-based CCFD algorithms, encompassing architectures such as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Autoencoders (AE), Deep Belief Networks (DBN), and hybrid models like CNN-RNN.

Recent advancements in fraud detection through machine learning techniques have markedly improved detection efficacy; nevertheless, centralized methodologies frequently provide severe privacy and regulatory problems, particularly in financial institutions where the safeguarding of sensitive data is crucial.

Federated learning (FL) has emerged as a significant option, enabling collaborative learning among distant financial institutions while preserving the confidentiality of private data [10] [11].

Recent research in federated learning for credit card fraud detection has investigated diverse deep learning architectures and privacy-preserving methodologies. Tang and Liu [12] introduced a Structured Data Transformer (SDT) model that incorporates federated learning, utilizing the self-attention mechanism of Transformers to adeptly capture intricate feature correlations in serialized transaction data while preserving data privacy among several banks. Their adaptive federated aggregation technique mitigates client heterogeneity and improves model convergence. Nonetheless, their methodology fails to include explicit oversampling approaches for addressing class imbalance, nor does it emphasize model explainability. Complementary to this, Liu et al. [13] employed a federated hybrid oversampling method using K-Means SMOTEENN, which combines K-Means clustering, Synthetic Minority Oversampling Technique (SMOTE), and Edited Nearest Neighbors (ENN) cleaning to effectively tackle class imbalance and noise in fraud detection datasets. Their approach partitions the data into clusters to better preserve local minority class distributions before applying oversampling and noise filtering, thereby enhancing model robustness and generalization. Integrated with a stacking ensemble of diverse classifiers, this method significantly improved detection performance on credit card fraud data, achieving superior precision, recall, and F1-score compared to traditional resampling techniques.

Meanwhile, A. M. Salih et al. [14] highlighted the

importance of Explainable AI techniques such as LIME and SHAP to provide transparency for deep learning models in regulated financial environments.

Our study builds upon these advances by integrating hybrid data balancing, federated learning, and explainability methods to achieve robust, interpretable fraud detection in distributed settings

III. Material and Method

A. Proposed Method

The workflow of the proposed method is illustrated in Fig. 1. Initially, raw data is collected and subjected to data preprocessing, which includes the application of the KMeans-SMOTEENN technique to handle imbalanced data effectively by generating synthetic samples and editing noisy instances. Afterwards, the federated learning (FL) framework is implemented, enabling collaborative training of deep learning models across distributed client environments without compromising data privacy. The deep learning models are then locally trained on each client's dataset and aggregated into a global model through the FL mechanism. Subsequently, evaluation metrics—including accuracy, precision, recall, F1-score, AUROC, and AUPRC—are computed to assess model performance. To enhance transparency and interpretability of the model decisions, the Explainable Artificial Intelligence (XAI) approach, specifically Local Interpretable Model-Agnostic Explanations (LIME), is integrated to explain the predictions and gain deeper insights into feature importance.

B. Dataset Description

This study employs a publicly accessible dataset of credit card transactions conducted by European cardholders. The European credit card fraud dataset was selected because it has become a widely accepted benchmark in fraud detection research, enabling direct and fair comparison with existing state-of-the-art methods. Its highly imbalanced distribution, with fraud cases accounting for only 0.172% of the total transactions, closely reflects real-world scenarios where fraudulent activities are rare but critical to detect.

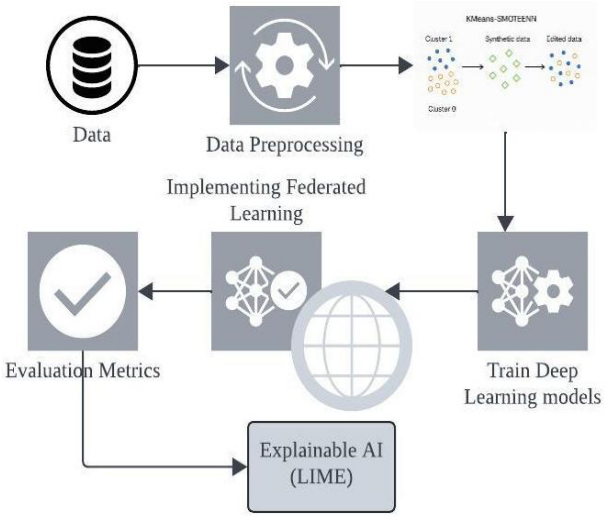


Fig. 1. Proposed method workflow.

The dataset's anonymized features, derived through PCA transformation, ensure compliance with data privacy and confidentiality requirements, aligning well with the regulatory considerations of financial applications. Furthermore, the dataset is publicly accessible, facilitating reproducibility and transparency in research. These characteristics make it an appropriate and representative choice for evaluating the effectiveness of federated learning in addressing fraud detection under severe class imbalance and privacy constraints.

The transactions encompass a two-day duration in September 2013, totaling 284,807 entries, of which merely 492 are classified as fraudulent. This indicates a significantly imbalanced class distribution, with fraudulent instances constituting merely 0.172% of the dataset. The dataset comprises 31 attributes: 'Time', 'Amount', 'Class', and the anonymized variables 'V1' to 'V28', which have been subjected to Principal Component Analysis (PCA) to maintain the confidentiality of sensitive data [1]. The dataset is suitable for machine learning operations because of its exclusively numeric properties. Moreover, its prevalent application in current research enables efficient benchmarking and performance evaluation against proven fraud detection algorithms.

This work involves data preprocessing in three primary stages: resampling to address class imbalance, standardization to equalize feature scales, and reshaping to prepare data for input into diverse machine learning architectures.

Fig. 2 demonstrates a significantly skewed class distribution within the dataset, comprising 283,253 normal transactions and merely 473 instances of fraud, or approximately 99.8% and 0.2% of the total data, respectively. This considerable disparity is a significant obstacle in the training of prediction models, as conventional classifiers often prioritize the majority class, hence risking the neglect of infrequent yet crucial fraudulent patterns. This discrepancy highlights the need for implementing resampling strategies to improve model sensitivity and guarantee effective fraud detection performance.

Considering that the minority class signifies fraudulent instances, employing the unaltered dataset without any balancing methods may considerably impair the model's capacity to identify fraud, resulting in suboptimal recall performance. An innovative hybrid resampling method, KMeans-SMOTEENN, is utilized to resolve this issue. This method integrates clustering-based oversampling with noise-filtering undersampling, facilitating a more efficient management of the dataset's class imbalance by producing synthetic minority samples while concurrently eliminating potential outliers and noisy majority samples.

Subsequent to resampling, feature standardization is executed utilizing StandardScaler, which is crucial owing to the diverse types and sizes of the dataset's features. Standardizing the data guarantees that each feature contributes uniformly to the learning process, hence improving model convergence and performance. Furthermore, the data is restructured by incorporating an additional dimension to align with the input specifications of the Neural Network model, which serves as the principal architecture for fraud detection in this research.

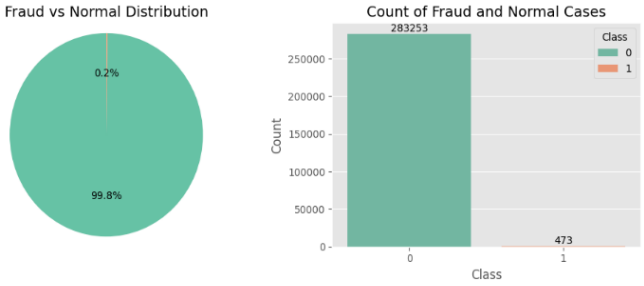


Fig. 2. Imbalance ratio of class fraud and non-fraud.

C. Deep Learning Model

This section succinctly elucidates the deep learning methodology employed in this work.

Since Hinton's breakthrough in 2006, deep learning has progressed swiftly, facilitating the development of robust architectures such as CNNs, RNNs, LSTMs, and Transformers. The breakthroughs, coupled with enhanced hardware capabilities and the accessibility of extensive open-source data, have established deep learning as a preeminent force in artificial intelligence [15].

The fundamental components of deep learning encompass backpropagation, stochastic gradient descent, and convolutional neural networks. It has proven useful in analyzing extensive datasets, especially in tasks such as semantic indexing, data annotation, and information retrieval. Nonetheless, obstacles persist, such as the management of high-dimensional and streaming data, in addition to guaranteeing model scalability. Notwithstanding these constraints, deep learning persists in its advancement, with continuous endeavors to create a more cohesive comprehension and methodology [16].

This research selected deep learning for its capacity to autonomously extract significant features from data, thus reducing the necessity for costly manual feature engineering. To examine the optimal architecture for resolving the research challenge, various deep learning models were deployed and assessed. These models typically consist of multi-layer neural networks, wherein each layer incrementally acquires data representations, spanning from fundamental properties to more intricate and abstract patterns.

In the studies, we employed six distinct variations of deep learning models to assess performance and identify the model yielding optimal results. The specifics of each model and the experimental findings will be comprehensively detailed in the experiment and analysis of results section. This methodology was employed to illustrate the generalizability of deep learning models and to identify the optimal solution depending on the dataset's specific attributes.

1) Autoencoder: Autoencoders are a particular category of neural networks intended to acquire efficient representations (encodings) of unlabeled input, primarily for dimensionality reduction or feature extraction [17].

2) FNN: A Feed-Forward Neural Network (FNN) is a category of artificial neural network in which information progresses unidirectionally from input, through hidden layers, to output without any cycles. Establishing the optimal network size, encompassing the quantity of hidden layers and neurons, is essential as it influences learning ability, generalization, and the likelihood of overfitting. The universal approximation theorem asserts that a single hidden-layer feedforward neural network can estimate any continuous function with a sufficient number of neurons; however, additional hidden layers frequently provide more efficient solutions [18].

3) CNN: A Convolutional Neural Network (CNN) is a prevalent deep learning method that has yielded favorable outcomes across several applications. Convolutional Networks can reveal hidden characteristics of fraudulent transactions and prevent model overfitting. The ConvNets algorithm comprises three primary layers: the convolution layer, the pooling layer, and the fully connected layer.

The convolution and pooling layers primarily execute feature extraction, but the fully connected layer subsequently maps the extracted features to the final output, such as classification [19].

4) LSTM: Long Short-Term Memory (LSTM) is a specialized architecture of artificial recurrent neural networks (RNN) employed to model time series data in deep learning. Unlike conventional feedforward neural networks, LSTM has feedback connections among hidden units linked to discrete time steps, enabling the learning of long-term sequence dependencies and the prediction of a transaction label based on the sequence of prior transactions [20].

5) DNN: A Deep Neural Network (DNN) is a kind of artificial neural network characterized by the presence of multiple hidden layers situated between the input and output layers, in contrast to shallow models. Like conventional neural networks, DNNs handle inputs by multiplying them with weights and transmitting the outcomes through hidden layers activated by nonlinear functions such as sigmoid, tanh, or ReLU. The model parameters are refined by minimizing an error function, typically by stochastic gradient descent, until convergence is achieved. DNN training has two primary phases utilizing the backpropagation algorithm: a forward pass, in which calculations flow from input to output, and a backward pass, during which incorrect gradients are propagated in reverse to adjust the weights[21].

6) STACKING-DL: Stacking is a significant ensemble learning strategy that synthesizes an optimal model by amalgamating predictions from many foundational machine learning algorithms in the initial layer (Emmanuel et al. 2023). The primary concept is to employ the prediction results of the base learner as input features, thereafter training and predicting with the meta-learner. During the model's training phase, an optimum combination of various machine learning methods is employed to fully leverage their respective strengths, hence enhancing the accuracy of the ensemble model's predictions [22].

Table I presents the configuration details and hyperparameters of the neural network models implemented in

this study for credit card fraud detection. The models include Feed-Forward Neural Network (FNN), Deep Neural Network (DNN), 1D Convolutional Neural Network (CNN 1D), Long Short-Term Memory (LSTM), Autoencoder, and a Hybrid model.

- Input Activation Function: All models utilize the Rectified Linear Unit (ReLU) activation function at the input or hidden layers to introduce non-linearity and facilitate efficient training.
- Output Activation Function: Sigmoid activation is applied in the output layer of all models except the Autoencoder, which uses a Sigmoid function suited for reconstruction tasks. This allows for output values between 0 and 1, suitable for binary classification problems like fraud detection.
- Optimizer: The Adam optimizer is consistently used across all models, with an initial learning rate primarily set at 0.001, except for CNN 1D, which uses a lower rate of 0.0001 to accommodate convolutional learning dynamics.
- Learning Rate Decay and Dropout: Learning rate decay and dropout regularization are employed selectively to improve generalization and prevent overfitting. DNN, CNN 1D, LSTM, and Hybrid models incorporate decay rates and dropout values ranging approximately between 0.2 and 0.5, while FNN and Autoencoder models omit these parameters.
- Communication Rounds and Federated Clients: All models are trained in a federated learning setup with 50 communication rounds and 2 to 3 federated clients participating in the training process, ensuring distributed learning and privacy preservation.
- Train-Test Split Ratio: A consistent 80%-10%-10% split is applied for training, validation, and testing datasets, respectively, to maintain fair performance evaluation across models.

This configuration setup balances model complexity and training efficiency, aiming to achieve optimal detection performance while maintaining computational feasibility in a federated environment.

TABLE I. MODELS HYPERPARAMETERS

Model Name	FNN	DNN	CNN 1D	LSTM	Autoencoder	Hybrid
Input Activation Function	ReLU	ReLU	ReLU	ReLU	ReLU	ReLU
Output Activation Function	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Sigmoid
Optimizer	Adam	Adam	Adam	Adam	Adam	Adam
Initial Learning Rate	0.001	0.001	0.0001	0.001	0.001	0.001
Learning rate decay	-	0.3	0.2	0.2	-	-
Dropout rate	-	0.3	0.2, 0.4	0.5	-	0.3, 0.2
Communication round	50	50	50	50	50	50
Number of federated clients	3	3	3	2	3	3
Train test ratio	80%-10%-10%	80%-10%-10%	80%-10%-10%	80%-10%-10%	80%-10%-10%	80%-10%-10%

D. K-Means SMOTEENN Imbalanced Handling Dataset

This work employs a hybrid data-balancing methodology that integrates K-Means clustering with SMOTE (Synthetic Minority Oversampling Technique) and ENN (Edited Nearest Neighbors), generally referred to as K-SMOTEENN. This strategy seeks to resolve issues of class imbalance and overlap between classes frequently encountered in datasets, particularly in the fraud detection dataset utilized.

The K-SMOTEENN method we utilize consists of three primary phases:

1) K-Means clustering: The K-Means approach is employed to partition the minority class data into multiple clusters according to feature similarities. Each cluster signifies a distinct pattern within the minority class that exhibits greater homogeneity than the class as a whole.

2) Oversampling utilizing SMOTE: Subsequent to the segmentation of minority data into multiple clusters, we implement the SMOTE technique on each cluster independently to produce supplementary synthetic samples. This procedure is conducted just on clusters that satisfy the minimal sample size criteria for efficacy.

3) Sanitation utilizing ENN: Upon completion of the oversampling phase, the resultant dataset from the oversampling is amalgamated with the majority class. The final stage involves employing the ENN approach to eliminate unclear or potentially noisy samples, hence enhancing the quality of the final data.

The integration of the three methodologies yields a more balanced dataset, facilitating the classification model's ability to differentiate samples from other classes, particularly in overlapping areas. Practical applications of this methodology utilize Python libraries, including scikit-learn for K-Means clustering and imblearn for oversampling and undersampling techniques such as SMOTE and ENN. Optimal configurations, including the number of K-Means clusters and sample parameters, are determined during first experimentation [13].

Thus, this enables K-SMOTEENN to focus on sample data; this strategy is delineated in Algorithm 1.

Algorithm 1 K-SMOTEENN [13]

Input: training data S is a set of pairs $\{(x1, y1), \ldots, (x2, y2), \ldots, (xm, ym)\}$ where *x* represents the input data and y is the corresponding target vector.
n (the number of samples)
k (indicates the number of clusters)
irt (imbalance ratio threshold)
knn (quantity of nearest neighbors)
begin
// Step 1: Divide the input space into clusters *clusters* ← *kmeans*(*X*)
filtered *clusters* ← empty set for c in cluster:

$$\text{imbalance ratio} \leftarrow \frac{\text{majority count(c)}+1}{\text{minority count (c)}+1}$$

If the imbalance ratio is less than the *irt*, add the cluster "c" to the filtered cluster set. Repeat this process until all clusters have been checked.
end
end
Step 2: Implement the SMOTE oversampling technique.
1) From the minority class, choose an instance x_i at random
2) Determine the kind of x_i and denote the samples as S_j
3) Randomly create a synthetic data point p by then selecting a sample in *Sj* called *z*, then creating a line segment in the feature space by connecting p and *z*
4) Minority class label assigned to *p*.
5) Create a series of synthetic instances by combining *p* and *z* convexly.
Step 3: Employing ENN techniques
1) Choose the arbitrary instance x_r from the set S
2) Determine the knn of x_r, with k being equal to 5
3) Remove the x_r Element if it has a more significant number of neighbors from the other class.
4) Iterate 6 to 8 steps for the entire training dataset.
End

E. Federated learning

Federated learning mainly refers to a distributed machine learning method implemented among numerous clients. The procedure involves N clients $\{C_1, C_2, ..., C_N\}$ indexed by k, each possessing its own local dataset $\{D_1, D_2, ..., D_N\}$, which is maintained locally, and data cannot be transferred between clients or gathered by a third party. Typically, a server organizes various clients and their training. FL encompasses three essential steps:

1) Initialization: At communication round *t*, the clients get the newest model wt from the server for initialization.

2) Local training: Each client C_k conducts iterative training based on its own local dataset D_k and hyperparameter η. The local model weight ω_t^k is updated to ω_{t+1}^k after certain training epochs according to $\omega_{t+1}^k \leftarrow \omega_t^k$ (η, $D_{k)}$, leftarrow and subsequently transmitted to the server.

3) Model aggregation: the server does model aggregation on the received local models and updates the global model $\omega_{glob}^{t+1} \leftarrow$ Agg $(\omega_t^{k+1}; k \in [1,....,N])$.

Thus, FL facilitates collaboration among different clients in training a model without the necessity of data exchange, which is particularly advantageous for privacy-sensitive applications [23].

This study utilizes the Federated Averaging (FedAvg) algorithm as the primary approach within our federated learning (FL) system. FedAvg is acknowledged as the predominant strategy in federated learning, enabling clients to collaboratively train a global model while refraining from sharing raw data. Each client develops a local model utilizing its own dataset, while the central server orchestrates parameter distribution, aggregation, and updates. The model parameters from all participants are averaged and redistributed until convergence is reached. This technique enhances data privacy, reduces communication overhead, and guarantees scalability for multiple customers [24]. The full procedure is depicted in Algorithm 2.

Algorithm 2 FedAvg [24]

1: **Server performs**:
2: k is indexed as the K clients,
3: the minibatch size is denoted by B,
4: the number of local epochs is E,
5: the learning rate is denoted as η
6: ω_0 is initialized by the server
7: **for** t =1, 2, … , T **do**
8: $m \leftarrow \max(C.K, 1)$
9: $S_t \leftarrow m$ clients (random set)
10: **for** $k \in S_t$ **do**
11: $\omega_{t+1}^k \leftarrow$ **ClientUpdate**(k,ω_t)
12: $\sum_{k=1}^{K} \frac{nk}{n} \omega_{t+1}^k$
13: **ClientUpdate**(k,ω):
14: $\beta \leftarrow$ split P_k into batches of size $\boldsymbol{B}$
15: **for** each local epoch i from 1 to E **do**
16: **for** batch $b \in \beta$ do
17: $\omega \leftarrow \omega - \eta \nabla l(\omega; b)$
18: **end for**
19**: end for**
20: **end for**
21: **end for**
22: return ω to server

Fig. 3 depicts the federated learning architecture utilized in this research. The system consists of a central server and numerous distributed clients (Client 1 to Client 10). Each client maintains a distinct local dataset, which is saved and processed privately without the dissemination of raw data. The central server first disseminates a global deep learning model to every client. Clients thereafter train their models locally with private datasets and transmit the updated model parameters to the server. The server consolidates these parameters through federated averaging or analogous aggregation techniques to enhance and update the global model. This iterative procedure persists until model convergence is attained, safeguarding data privacy, reducing communication overhead, and adeptly managing heterogeneous data distributions among clients.

IV. Result and Discussion

This research presents an innovative approach to credit card fraud detection by leveraging federated learning (FL) combined with deep neural network models. To address the prevalent data imbalance issue in fraud detection, we employed the KMeans-SMOTEENN resampling technique during preprocessing. The experimental results are structured into two main sections:

metrics comparison per training round and global (overall) metrics across all rounds. The federated learning framework incorporated six distinct deep learning architectures: Autoencoder, CNN, DNN, FNN, LSTM, and a stacked deep learning model. We implemented the FL process using the Flower federated learning framework, a widely recognized Python library tailored for distributed learning scenarios.

All experiments were conducted on a Windows-based workstation, powered by an Intel(R) Core(TM) i7-10700 CPU @ 2.90GHz, and equipped with 16 GB RAM. Model performance was rigorously evaluated using multiple performance indicators: Accuracy, Precision, Recall, F1-score, AUROC (Area Under the ROC Curve), and AUPRC (Area Under the Precision-Recall Curve). The ROC curve graphically represents the classifier's capability in discriminating between fraudulent and non-fraudulent transactions by plotting the True Positive Rate against the False Positive Rate at varying thresholds. AUROC provides a succinct numerical summary of this capability, ranging from 0 to 1, where a value close to 1 signifies excellent predictive accuracy. Additionally, the Precision-Recall Curve focuses specifically on the performance of the classifier in imbalanced class scenarios, such as fraud detection tasks. The subsequent subsections provide detailed visualizations and analyses of the per-round and global performance metrics for each deep learning architecture utilized in the federated learning framework.

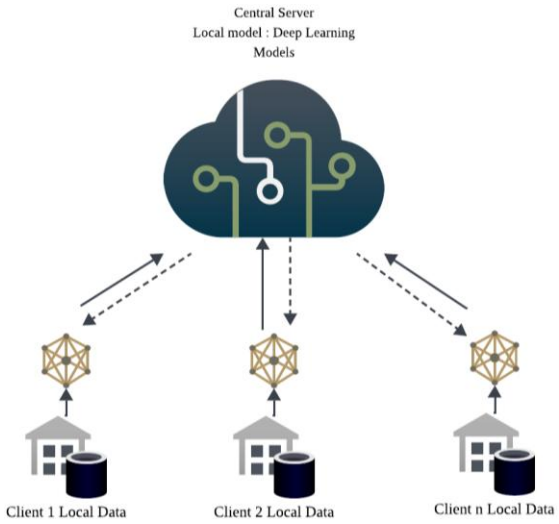


Fig. 3. Federated learning architecture.

A. Performance Evaluation

This section presents the assessment of our proposed federated learning framework for multi-client credit card fraud detection. To thoroughly evaluate the efficacy of various deep learning architectures employed in this federated framework, we analyzed their global performance indicators, consolidated over all clients and training iterations. The assessed designs are Autoencoder, Convolutional Neural Network (CNN), Deep Neural Network (DNN), Feedforward Neural Network (FNN), Long Short-Term Memory (LSTM), and Stacked Deep Learning model.

Table II delineates the comparative performance metrics—Accuracy, Precision, Recall, F1-score, AUROC (Area Under the Receiver Operating Characteristic Curve), and AUPRC (Area Under the Precision-Recall Curve)—for each deep learning model. All models demonstrated consistently high scores across the assessed metrics. The accuracy varied from 0.9992 to 0.9993, signifying an exceptionally high rate of right classifications between fraudulent and legitimate transactions. Precision scores were consistently outstanding, ranging from 0.9993 to 0.9994, indicating the models' exceptional proficiency in accurately recognizing actual positive cases, hence reducing false positive detections.

Recall metrics, which assess the models' efficacy in detecting genuine fraudulent cases (true positive rate), consistently exhibited elevated values (ranging from 0.9992 to 0.9993). Correspondingly, the F1-scores—which equilibrate precision and recall—were remarkably elevated (about 0.9992 to 0.9993), highlighting the models' equitable performance and formidable predictive capability in identifying fraudulent behaviors inside the federated learning framework.

To further assess the model's capacity to differentiate between classes, we used the AUROC and AUPRC measures. The AUROC scores for all models varied between 0.9748 and 0.9833, indicating that all models possess exceptional discriminative ability in differentiating fraudulent transactions from legitimate ones. Nevertheless, there was modest heterogeneity in the AUPRC ratings, indicating slight discrepancies in the efficacy of each model in managing the intrinsically imbalanced fraud detection data. The DNN got the highest AUPRC score (0.8420), closely succeeded by layered deep learning (0.8383) and CNN (0.8289), underscoring these models' specific capabilities in precision-recall equilibrium, a vital component in fraud detection contexts characterized by a scarcity of positive class cases.

TABLE II. PERFORMANCE COMPARISON OF EACH MODEL

Model Type	Accuracy	Precision	Recall	F1	Auroc	Auprc
Autoencoder	0.9993	0.9994	0.9993	0.9993	0.9748	0.8119
Cnn	0.9993	0.9993	0.9993	0.9993	0.9816	0.8289
Dnn	0.9993	0.9993	0.9993	0.9993	0.9833	0.8420
Fnn	0.9992	0.9993	0.9992	0.9992	0.9798	0.8117
Lstm	0.9993	0.9993	0.9993	0.9992	0.9708	0.8058
Stacked_dl	0.9993	0.9993	0.9993	0.9993	0.9855	0.8383

In addition to the tabular overview, Fig. 5 visually illustrates the comparative global metrics among the six architectures. The illustrated bar graphs validate the superior and closely comparable performance of each model across all measures, underscoring the resilience and dependability of our federated learning methodology. This consistency indicates that federated learning, when combined with suitable data resampling methods (namely KMeans-SMOTEENN) and diverse neural network topologies, may proficiently address data imbalance, resulting in highly accurate and balanced predicted outcomes.

In conclusion, both the table and graphical analyses unequivocally endorse the efficacy and dependability of our federated neural network methodology in identifying credit card fraud across various clients, yielding robust, equitable, and interpretable predicted results.

B. *Explainable AI with LIME*

We employed Explainable AI through the LIME (Local Interpretable Model-agnostic Explanations) technique to enhance the interpretability of the Deep Neural Network (DNN) model's predictions in fraud detection. Fig. 4 demonstrates LIME visualization for a transaction classified as non-fraudulent with complete certainty (probability = 1.00) that features V11, V2, and V17 significantly bolster the non-fraud forecast (shown in blue), but features V14, V3, and V15 marginally suggest a propensity for fraud (marked in orange).

Despite certain indicators indicating possible fraud, their total impact is negligible, therefore corroborating the model's determination as non-fraudulent. This visualization improves transparency in the model's decision-making process and elucidates the significance of various attributes in certain forecasts, which is essential for stakeholder comprehension in financial contexts.

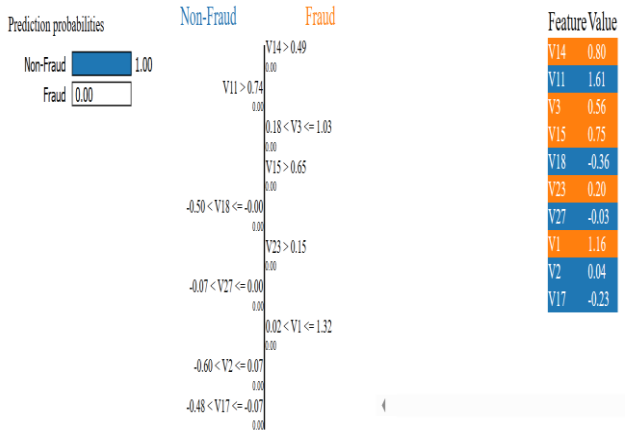


Fig. 4. Explainable AI.

C. *Performance Comparison with State-of-the-Art Methods*

To assess the efficacy and originality of our proposed approach, we performed a comparison analysis using other cutting-edge federated learning and unbalanced data management strategies documented in the literature. Table III demonstrates that our proposed model, which employs a Deep Neural Network (DNN) combined with the K-Means SMOTEENN method, exhibits exceptional performance, surpassing the majority of existing methodologies across various evaluation metrics, including Accuracy, Recall, Precision, F1-score, AUC, and AUPRC. Our method exhibited exceptional precision, recall, and F1-score, each at 99.93%, in addition to a high AUC value of 98.33% and a significantly raised AUPRC of 84.20%, outperforming comparable contemporary methodologies including those of Mustafa et al. [6], Liu et al. [13], and Saha et al. [29]. This thorough performance comparison highlights the efficacy and innovation of our federated learning methodology, especially in tackling the issues of imbalanced datasets in credit card fraud detection tasks.

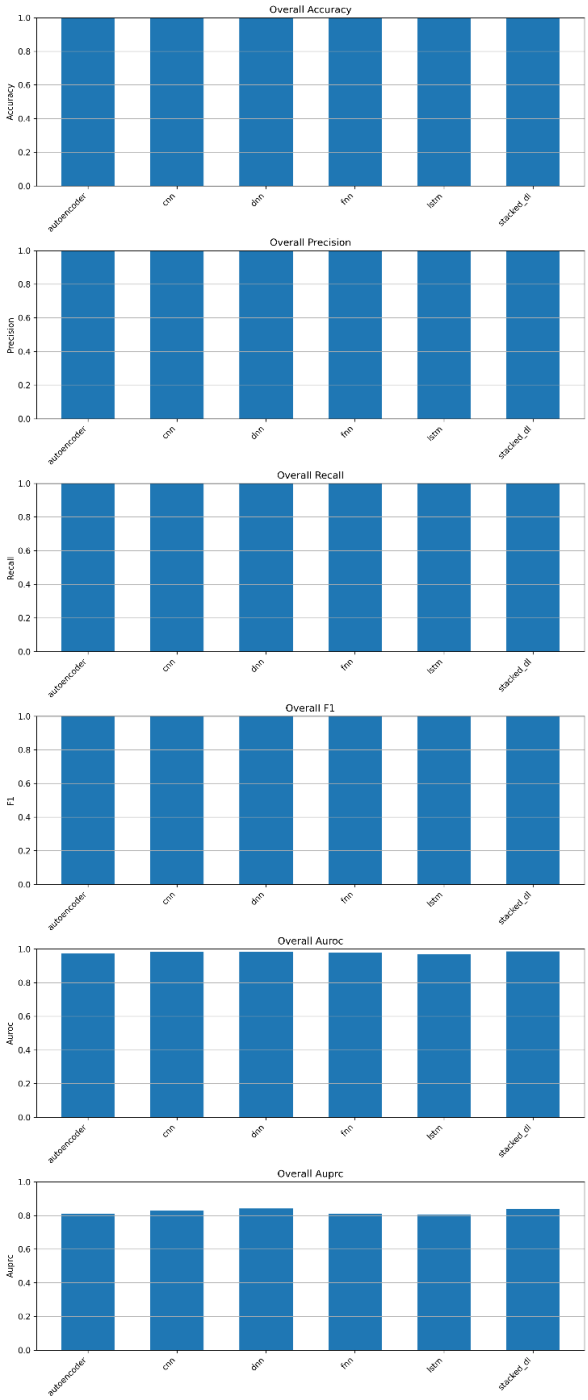


Fig. 5. Performance evaluation.

D. Discussion

Our findings illustrate the efficacy of integrating federated learning with the FedAvg algorithm, a K-Means-SMOTEENN hybrid methodology, and a Deep Neural Network (DNN) in tackling significant obstacles in fraud detection across several customers. The implementation of federated learning (FL) markedly improves data privacy by facilitating collaborative model training without the exchange of sensitive client information, consistent with prior study findings [1, 2]. Furthermore, the FedAvg technique effectively consolidates local models, yielding consistently superior prediction performance across diverse client datasets. The use of KMeans-SMOTEENN significantly enhanced the management of the pronounced class imbalance characteristic of fraud detection datasets. This hybrid approach substantially reduced the overlapping distribution between fraudulent and non-fraudulent occurrences by producing representative synthetic samples inside confined clusters. Our findings validate the results of other studies [3, 4], demonstrating that cluster-based oversampling strategies surpass conventional methods in intricate, imbalanced situations.

Additionally, we utilized Explainable AI (XAI) methodologies, specifically the LIME approach, to elucidate the predictions generated by our DNN model. The LIME display emphasized the most significant features impacting individual predictions, enhancing model transparency and interpretability. The interpretability of automated judgments is essential for stakeholders, particularly in financial services, since it enhances confidence and ensures regulatory compliance, as highlighted by Ribeiro et al. [5].

Notwithstanding the attainment of encouraging outcomes, our study possesses specific limitations. The existing solution predominantly assesses federated learning in simulated client environments, which may not accurately reflect real-world operational conditions characterized by network latency, data heterogeneity, and diverse compute resources among clients. Moreover, the utilization of synthetic data, although efficient, may generate artifacts that inadequately reflect authentic fraud trends.

TABLE III. PERFORMANCE COMPARISON WITH STATE-OF-THE-ART METHODS

Ref	Year	Method Use	Imbalanced Handling	Accuracy(%)	Recall	Precision	F1 Score	AUC	AUPPRC
Yang et al.[25]	2019	FL with CNN	-	-	-	-	-	-	-
Suvarna et al. [26]	2020	FL with RBF	-	94.00	-	-	-	-	-
Forough et al. [27]	2021	LSTM	-	-	74.08	95.6	78.1	83.3	-
Aurna et al. [28]	2023	FL with CNN	SMOTE	99.11	99.51	98.71	99.11	-	-
Aurna et al. [28]	2023	FL with MLP	SMOTE	98.28	98.77	97.78	98.28	-	-
Aurna et al. [28]	2023	FL with LSTM	SMOTE	95.78	98.20	80.23	88.31	-	-
Yuxuan et al. [12]	2024	Federated SDT	-	99.8	79.5	85.0	82.2	99.8	89.2
Mustafa et al. [6]	2024	Federated + CNN	SMOTE	-	80.9	8.26	81.7	93.7	-
Liu et al. [13]	2025	Stacking Ensemble without FL	K-Means SMOTEENN	1.00	0.88	0.95	0.92	1.00	0.96
SC Saha et al. [29]	2025	FinGraphFL	-	0.9780	-	-	-	0.9670	-
Proposed method	**2025**	**Our best model (DNN)**	**K-Means SMOTEENN**	**99.93**	**99.93**	**99.93**	**99.93**	**98.33**	**84.20**

V. CONCLUSION

This study introduces an innovative methodology that combines federated learning using FedAvg, KMeans-SMOTEENN to tackle data imbalance, and Explainable AI with a Deep Neural Network model to improve multi-client fraud detection systems. Our proposed method exhibited strong predictive performance, enhanced data privacy protection, effectively tackled class imbalance issues, and provided clear model interpretability.

The integration of these methodologies enhanced stakeholder trust and fostered a more profound comprehension of model-driven decisions in the financial sector.

Future research has found numerous promising avenues. Investigating sophisticated federated learning methodologies, including personalized FL (e.g., FedProx, FedBABU, or adaptive FL approaches), may enhance model efficacy and adaptability to client-specific data distributions. Moreover, the integration of Differential Privacy approaches would augment data privacy, providing enhanced safeguards against potential inference assaults.

Finally, applying this methodology to extensive real-world deployments, including dynamic and streaming data, could assess and enhance the practical usability and resilience of our suggested strategy.

ACKNOWLEDGMENT

This work was supported by the National Science and Technology Council (NSTC), Taiwan, with the grants 113-2221-E-027-051 and 114-2221-E-027-067.

REFERENCES

[1] V. V. K. Reddy, R. V. K. Reddy, M. S. K. Munaga, B. Karnam, S. K. Maddila, and C. S. Kolli, "Deep learning-based credit card fraud detection in federated learning," Expert Syst. Appl., vol. 255, p. 124493, 2024.

[2] M. Adil, Z. Yinjun, M. M. Jamjoom, and Z. Ullah, "OptDevNet: A Optimized Deep Event-based Network Framework for Credit Card Fraud Detection," IEEE Access, 2024.

[3] M. F. Ahamed, A. Salam, M. Nahiduzzaman, M. Abdullah-Al-Wadud, and S. M. R. Islam, "Streamlining plant disease diagnosis with convolutional neural networks and edge devices," Neural Comput. Appl., vol. 36, no. 29, pp. 18445–18477, 2024.

[4] S. K. Aljunaid, S. J. Almheiri, H. Dawood, and M. A. Khan, "Secure and transparent banking: explainable AI-driven federated learning model for financial fraud detection," J. Risk Financ. Manag., vol. 18, no. 4, p. 179, 2025.

[5] S. Ji et al., "Emerging trends in federated learning: From model fusion to federated x learning," Int. J. Mach. Learn. Cybern., vol. 15, no. 9, pp. 3769–3790, 2024.

[6] M. Abdul Salam, K. M. Fouad, D. L. Elbably, and S. M. Elsayed, "Federated learning model for credit card fraud detection with data balancing techniques," Neural Comput. Appl., vol. 36, no. 11, pp. 6231–6256, 2024.

[7] A. R. Khalid, N. Owoh, O. Uthmani, M. Ashawa, J. Osamor, and J. Adejoh, "Enhancing credit card fraud detection: an ensemble machine learning approach," Big Data Cogn. Comput., vol. 8, no. 1, p. 6, 2024.

[8] I. D. Mienye and N. Jere, "Deep learning for credit card fraud detection: A review of algorithms, challenges, and solutions," IEEE Access, 2024.

[9] X. Feng and S.-K. Kim, "Novel machine learning based credit card fraud detection systems," Mathematics, vol. 12, no. 12, p. 1869, 2024.

[10] M. A. Salam, D. L. El-Bably, K. M. Fouad, and M. Elsayed, "Enhancing Fraud Detection in Credit Card Transactions using Optimized Federated learning Model.," Int. J. Adv. Comput. Sci. Appl., vol. 15, no. 5, 2024.

[11] T. Baabdullah, A. Alzahrani, D. B. Rawat, and C. Liu, "Efficiency of federated learning and blockchain in preserving privacy and enhancing the performance of credit card fraud detection (CCFD) systems," Futur. Internet, vol. 16, no. 6, p. 196, 2024.

[12] Y. Tang and Z. Liu, "A Credit Card Fraud Detection Algorithm Based on SDT and Federated learning," IEEE Access, vol. 12, pp. 182547–182560, 2024.

[13] N. Damanik and C.-M. Liu, "Advanced Fraud Detection: Leveraging K-SMOTEENN and Stacking Ensemble to Tackle Data Imbalance and Extract Insights," IEEE Access, 2025.

[14] A. M. Salih et al., "A perspective on explainable artificial intelligence methods: SHAP and LIME," Adv. Intell. Syst., vol. 7, no. 1, p. 2400304, 2025.

[15] X. Chen et al., "Deep learning-based software engineering: progress, challenges, and opportunities," Sci. China Inf. Sci., vol. 68, no. 1, pp. 1–88, 2025.

[16] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015.

[17] I. D. Mienye and T. G. Swart, "Deep autoencoder neural networks: A comprehensive review and new perspectives," Arch. Comput. methods Eng., pp. 1–20, 2025.

[18] G. Bebis and M. Georgiopoulos, "Feed-forward neural networks," Ieee Potentials, vol. 13, no. 4, pp. 27–31, 1994.

[19] M. L. Gambo, A. Zainal, and M. N. Kassim, "A convolutional neural network model for credit card fraud detection," in 2022 International Conference on Data Science and Its Applications (ICoDSA), 2022, pp. 198–202.

[20] I. Benchaji, S. Douzi, B. El Ouahidi, and J. Jaafari, "Enhanced credit card fraud detection based on attention mechanism and LSTM deep model," J. Big Data, vol. 8, no. 1, p. 151, 2021.

[21] N. Nguyen et al., "A proposed model for card fraud detection based on Catboost and deep neural network," Ieee Access, vol. 10, pp. 96852–96861, 2022.

[22] W. Wang, M. Gu, Z. Li, Y. Hong, H. Zang, and D. Xu, "A stacking ensemble machine learning model for improving monthly runoff prediction," Earth Sci. Informatics, vol. 18, no. 1, p. 120, 2025.

[23] P. Qi, D. Chiaro, A. Guzzo, M. Ianni, G. Fortino, and F. Piccialli, "Model aggregation techniques in federated learning: A comprehensive survey," Futur. Gener. Comput. Syst., vol. 150, pp. 272–293, 2024.

[24] B. Yurdem, M. Kuzlu, M. K. Gullu, F. O. Catak, and M. Tabassum, "Federated learning: Overview, strategies, applications, tools and future directions," Heliyon, vol. 10, no. 19, 2024.

[25] W. Yang, Y. Zhang, K. Ye, L. Li, and C.-Z. Xu, "Ffd: A federated learning based method for credit card fraud detection," in International conference on big data, 2019, pp. 18–32.

[26] R. Suvarna and A. M. Kowshalya, "Credit card fraud detection using federated learning techniques," Int. J. Sci. Res. Sci. Eng. Technol., vol. 7, no. 3, 2020.

[27] J. Forough and S. Momtazi, "Ensemble of deep sequential models for credit card fraud detection," Appl. Soft Comput., vol. 99, p. 106883, 2021.

[28] N. F. Aurna, M. D. Hossain, Y. Taenaka, and Y. Kadobayashi, "Federated learning-based credit card fraud detection: Performance analysis with sampling methods and deep learning algorithms," in 2023 IEEE International Conference on Cyber Security and Resilience (CSR), 2023, pp. 180–186.

[29] Z. Xia and S. C. Saha, "FinGraphFL: Financial Graph-Based Federated learning for Enhanced Credit Card Fraud Detection," Mathematics, vol. 13, no. 9, p. 1396, 2025.

A Novel Multilevel Framework for DoS Detection in SDN

Rejo Rajan Mathew[1], Amarsinh Vidhate[2]
Department of Information Technology, Ramrao Adik Institute of Technology, D Y Patil University, India[1]
Department of Information Technology, MPSTME, SVKMs NMIMS University, India[1]
Department of Computer Engineering, Ramrao Adik Institute of Technology, D Y Patil University, India[2]

***Abstract*—DoS attacks have been the most popular type of attack on SDNs. The threat landscape has widened due to advanced persistent threats. Recent studies have focused on a single level of defence and conventional detection methods, which have become redundant. The study proposes and implements a novel multilevel DoS attack detection, which has a three-pronged approach to counter modern-day DoS attacks. The first level emphasizes the Zero Trust mechanism using Hash SHA-256 to validate the clients. The second level uses hybrid deep learning models to detect DoS attacks, which are trained and tested across three latest datasets, namely NSLKDD, CIC DOS 2019 and IOT2023, giving an accuracy of 95% consistently. The third level is a lightweight adaptive DoS detection, which can detect fast and low-rate DoS attacks, ensuring that the SDN is secure in a few milliseconds by ruling out any possibility of congestion. The results clearly indicate how a three-level approach can thwart most advanced persistent threats.**

***Keywords*—Software defined network; distributed denial of service; openflow**

I. INTRODUCTION

Traditional networks are decentralized, rigid to allow changes, and complex to configure and manage due to vendor-specific issues. SDN is a disruptive innovation as it allows remote management of large-scale deployments with ease and takes care of the rising demand for processing, storage, and networking. It has a centralized and programmable controller along with switches in the data plane. SDN takes advantage of virtualization to provide flexibility, adaptability, elasticity, scalability, robustness, security, and abstraction of the hardware [1]. Below mentioned are the vulnerabilities in SDN based on layers:

A. Control Layer (Control Plane)

As the central element of SDN, if the controller is not managed correctly, it could be a single point of vulnerability and attack. The controller establishes the direction of data flow within the data plane. When the controller is attacked and compromised, it will have a negative impact on the network.

1) Hijacked rogue controller: The attacker takes control of the original controller or changes its functionality such that they can play around with the network. Some attackers replace the actual controller with a rogue controller, which controls the entire network thereafter. [2]

2) Fake traffic and network manipulation: Intruder exhausts all the resources by introducing fake packets and simultaneously initiates other attacks on the entire network. [3, 4, 5]

3) Weak authentication, authorization and information disclosure: When the controller has weak authentication and incomplete encryption, the attacker breaches confidentiality, leading to information disclosure, making it susceptible [6].

B. Infrastructure Layer (Data Plane)

Flow Switches are non-intelligent and can be manipulated by attackers [7, 8].

1) Flow table modification: The malicious nodes alter the flow table rules by adding or deleting rules and denying updates to the table.

2) Flow table overloading/buffer saturation: Flow table rules are stored in TCAM, which is expensive, power hungry, with limited space, making it vulnerable to overloading and buffer saturation attacks. They can jam the bandwidth to cause overloading.

3) Topology spoofing/traffic diversion: Create a fake network view to redirect traffic flows to attackers and explore vulnerabilities such as eavesdropping. They can forge non-existent links between switches.

4) Side channel attack: It targets the timing information by collecting ping timings and adding delays. Also, it can inform an attacker if a flow rule exists or not.

C. Application Layer (Application Plane)

Some contemporary attackers have started to exploit the applications present in the application layer [9].

1) Malicious applications: Attacker can modify or run malicious applications to ruin the network or collect information of the network.

2) Resource exhaustion: Attacker can run resource intensive requests to controller bringing down the system.

3) Policy conflicts: Applications running on single controller give conflicting policy instructions leading to delay and confusion.

4) Privilege elevation: One application gains higher privilege access over other applications and later controls and manipulates them.

D. Interfaces

1) Northbound interface: It serves as the application layer's and the control layer's open, local, non-standard, and non-encrypted connection interface. It collects network details, state resource requirements from the upper layer and gives directions to the controller in the control plane. As there are various types of REST APIs used here, this becomes a huge point of vulnerability for attackers to tamper or eavesdrop [10].

2) Southbound interface: It serves as the control layer's and data layer's open, local, and non-encrypted connection interface. The controller uses it to create and modify rules in the Flow table. Rules can be intercepted and manipulated due to a lack of security [11]. TLS security is recommended but not mandatory, making it vulnerable to contemporary attacks [12].

3) East-West bound interface: In most architectures, there is a single controller, and there are distributed multi-controller architectures to manage the SDN. The interface between these controllers is called the East-West bound interface. These controllers synchronise and work with each other for high availability by exchanging notifications. They share updates about the network, like the state, devices, changes in the flow table, list of hosts, etc. Due to a lack of security, attackers compromise and exploit the controller [13] interface between the switches. In the data plane, switches are interconnected via links on which packets traverse till they reach the destination. Links are open, local and not encoded, making it vulnerable.

Section II discusses the various DDoS defense approaches in related work. Section III presents the proposed method. Section IV highlights the experimental setup and results. Section V presents the conclusion.

II. Related Work

DDoS defense approaches are classified into two categories based on where these defense models have been deployed. Most attacks are carried out on the Controller rather than the switch.

A. Controller-Based Solutions

Bringing down the controller helps destroy the entire SDN. The main target is the controller, and the only way to protect the entire SDN is by protecting the controller. Over the years, many solutions have been proposed by researchers to protect the controller. Four major categories are: one is entropy-based, Machine learning, Neural Networks, Statistical analysis-based.

1) Information Entropy: During normal days, the traffic flow is normal, and the entropy value is maximum. As soon as the attack starts, the traffic flow increases and entropy starts to decrease. The threshold level for entropy is set. If the traffic flow goes below this level, then it triggers the detection mechanism. In [14], the authors compute the threshold value five times, and then it triggers the detection mechanism. In [15], there is a two-step mechanism, in the first step, it checks the decrease in the threshold value, and in the second step, if the value decreases, it checks MAC addresses to differentiate between flash traffic and attack traffic. There are two modules in [16], the first one is for detection based on the threshold value, and the second module is used for mitigation by blocking unacknowledged SYN packets from the client. In [17], there are three phases, the first is normal phase of counting of packets, second is the preparatory phase, if value goes below the threshold, third is mitigate phase by dropping packets. Authors in [18] built a trust framework to identify malicious flow based on Kullback–Leibler divergence or relative entropy method. In [19], the authors use Jensen-Renyi divergence measure to estimate the variation from the threshold value for normal as well as malicious traffic. This can detect both high and low-rate attacks. Authors in [20] use three values- counter, entropy, and traffic rate. For detection as the packet flow increases the entropy decreases count gets incremented. For mitigation, it collects DPID, port and IP information to block the malicious packets.

2) Machine learning: Based on historical data or changes in traffic flow, the system learns and detects the presence of DDoS attacks. Authors [21] propose aggregating packets to create signatures which can be later used to detect attacks. In [22] authors use a sequence of feature values to predict and detect attacks. Authors in [23] classify attacks using machine learning and neural networks. In [24], an SVM classifier was used after the authors extracted key features. In [25], the authors solved the multi-class classification problem by advancing the SVM algorithms. In [26], the authors implemented and compared MLP, Decision Tree, Random Forest and SVM to conclude that Decision Tree gives the best results. Authors in [27] extracted the features from the traffic to draw the feature pattern graph. In [34], the authors used entropy on ports and applied the KNN algorithm to detect and drop malicious traffic. In [28], the authors worked on KNN to get better accuracy. In [29], the authors classify the traffic based on packet flow, size and speed. In [30], the authors figured out the attack's intensity and classified the traffic using a KNN classifier. In [31], the authors implemented the Factorization Machine algorithm to mitigate low-rate DDoS attacks. In [32], the authors developed a multi-plane security framework named VARMAN, where the switch is modified to detect anomalies and then apply ML-based classifiers to detect attacks. In [33], the authors worked on programmable switches to improve efficiency. In [34], the authors deployed a trigger mechanism on the switches which alerts the controller, who activates the ML classifier (KNN and K-means) to detect the malicious traffic. In [35], the authors implemented a low-volume defense solution with the following models (MLP, Random Forest, Decision Tree, SVM, REP Tree, J48). In [36], the authors implemented a hybrid ML algorithm (Support Vector Classifier and Random Forest) to detect malicious traffic. In [37], the authors implemented the Redis Simple Message Queue (RSMQ) mechanism with the following machine learning models (ExtraTrees, Decision Tree, KNN and Naive Bayes) to improve the accuracy of detection. In

[38], the authors classified traffic using following machine learning models (MLP, Random Forest, Logistic Regression, DT and AdaBoost).

3) Artificial neural networks: Authors in [39] implement a mechanism which traces the source of the attack as it maintains the details of the entire network and a mitigation module to delete the malicious table flow entries. Authors in [40] classify the traffic using fuzzy logic combined with L1-Extreme Learning Machines. By employing both unsupervised hashing and the discrete-time finite-state Markov chain (DTMC) model, it mitigates attacks. In [41], the controller acts selectively and focuses only on affected switches by applying the fuzzy c-means algorithm.

4) Statistical analysis: In [42], the source is verified before accepting packets. The controller acts as a proxy and verifies the source, and sends SYNACK. In [43], it maintains the state and connection list of all TCPs. This list keeps on increasing till it reaches the set limit. Once it crosses that limit, the controller blocks packets from those sources. The authors designed the RADAR framework in which the controller selects certain flows to be investigated. Authors too confirm the legitimacy of the source, and the algorithm verifies the information in the Packet-In message with this dictionary of Port, IP, and MAC addresses. The malicious traffic is taken care of by a backup controller, which temporarily saves these requests as the attack begins, then quickly sends them to the main controller. Authors compare the threshold value to the quantity of Packet-In messages with a certain header received during a specific time interval. If it crosses the predetermined value, all packets are deep scanned.

B. Switch-Based Solutions

1) TCP proxies: LineSwitch deploys proxies based on random distribution and ensures blocking malicious traffic from reaching the control plane. In another paper, both act as proxy and minimizes the timeout to improve the efficiency. Authors provide multi-layer protection by implementing filters at all entry and exit points on the switches.

2) Information entropy: Implements the algorithm on the switches instead of overloading the controller. Authors deep-scan the packets for spoofing/changes, and once it locates the attacker host, it changes the table rules to block this host. To identify an attack, it also computes entropy within a predetermined time interval. The programmable switches cannot decode the arithmetic logarithmic function, so to calculate entropy, the authors utilize the longest-prefix match (LPM) table. It determines the IP-based entropy at the source and destination. Authors detect an attack on programmable switches by computing the entropy for IP addresses. They use the longest-prefix match (LPM) table and bespoke count-sketches, respectively, to approximate frequencies and carry out computationally complex mathematical operations.

3) Artificial neural networks: Used switches as neurons in a distributed intrusion prevention system based on BPNN. Parallel operation is used by switches on the same layer. For easy computations, each swap uses minimal resources. Further, they deployed a Radial Basis Function (RBF) neural network to enhance this work's accuracy. Authors propose a simple DDoS mechanisms having machine learning based, traditional IDS on a heterogeneous system including FPGA boards, GPU, and host processors.

4) Flow statistics: The security tools such a DNS reflection defense, firewall and extensive filtering have been installed. In [14], the authors, to counteract HTTP GET flooding assaults, per-URL counting technique was created on NetFPGA-based Openflow switches. On the programmable switches, the DDoS detection logic was deployed. The switch computes a hash value using the source's MAC and IP addresses and compares it to one that was already computed.

5) Smart switches: Authors deployed smart switches which would perform source address matching. If there is a mismatch it will automatically alert and block the traffic. It helps in early detection but will lead to overloading the controller with invalid requests. In another paper the source packets are filtered, and genuine packet traffic is given specific score so as to be allowed. This method is useful to limit the attack, but if the attacker bypasses this filter, it will fail. Authors use a port-hopping defense mechanism, where the target is moved to the controller, reducing the chances of an insider attack. In this, we need to add more moving target defense mechanisms, such as address hopping, service hopping, path hopping, etc.

III. Proposed Method

The traffic flow consists of information from regular clients as well as the DoS Attacker. The proposed method shown in Fig. 1 comprises three levels to ensure robust DoS attack detection:

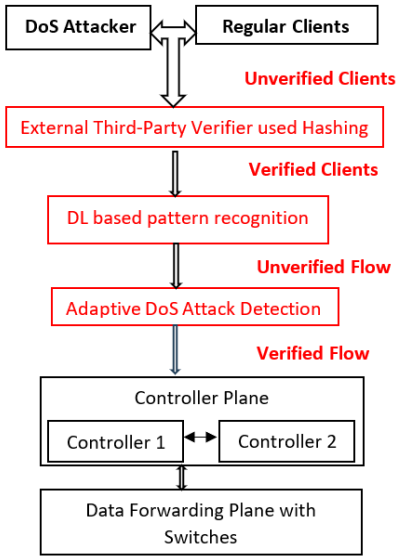


Fig. 1. Proposed method.

First Level – The first level allows only regular flow by matching the hash of the current flow with the hash of the

previously stored record. This ensures that only the verified clients go ahead to the next level.

Second Level – It consists of a deep learning-based hybrid model, namely RNN + LSTM, which has been trained and tested across three datasets to classify DoS attacks based on flow and performance-based parameters.

Third Level – Most APTs these days can mix Fast as well as Low-rate DoS Attacks to bypass the defense systems. So, we deploy the Adaptive DoS Attack detection, which will check all parameters and ensure only verified flows reach the controller.

A. Client Validation – Level 1

Fig. 2 shows an external third-party component that performs this client validation to ensure authorized users move on to the next level and reach the controller.

1) Collect - MAC Address and Source Port Number from each flow is collected by the third-party verifier.

2) Convert and Store – Third-party verifier converts the collected data into a hash value (SHA-256) and stores it in a hash table. The hash table consists of the hash value of known/regular clients.

3) Match - Repeat Step 1 and Step 2 to match the hash value of the current flow with the hash value of the flows stored in the table. If it matches, it will be classified as normal traffic and move to the next level or else block and do not allow that client to move ahead.

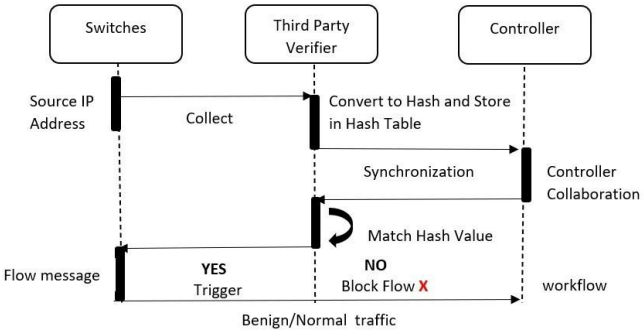


Fig. 2. Client validation.

B. Deep Learning Based DoS Detection – Level 2

The process of creating a Deep Learning based DoS attack detection system employs a methodical process beginning with the selection of three popular datasets. These datasets are NSL-KDD, CIC-DDoS 2019, and IoT-2023, which are intended to cover different types of network traffic patterns as well as possible attack types. To better illustrate these attack types, the datasets were focused on these specific attacks because they represent very prevalent attack types in real-life conditions. Certain attacks including SYN flood, UDP flood, and SQL Injection, were narrowed down for training of the model with intent of distinguishing between DoS/DDoS attacks and other attacks examples being ARP spoofing and command injection attacks. During model implementation, experimentation was carried on several deep learning architectures, namely the RNN, LSTM networks and GRU. These models were selected due to their suitability to time series data as well as their capabilities of recognizing sequential patterns in the network traffic, with LSTM networks being the superior of all in learning long-term dependencies of the data. Among the reasons for adopting such models in this work is the affordability of the economies delivered. That's the sheer speed in real-time attack detection that the models are known to allow. Some models that incorporated RNN, LSTM and GRU were also trained. Due to phased traffic across complex systems aimed at enhancing detection rates. These models were learnt in interludes in order to save time, along with hyperparameter tuning to manage factors such as learning rates, batch sizes, and number of epochs, among others, with the aim of improving model accuracy and reducing overfitting. The models achieved an assessment based on vital performance indicators, specifically accuracy, precision, recall, and the f1 score. Accuracy is the ratio of the correct predictions made to the total amount of predicted traffic given. Precision is the ratio of the correctly predicted attack traffic to the overall predicted attack traffic. Recall is the ratio of the predicted attacks captured relative to the total predicted attacks. The F1-score is the harmonic mean of precision and recall. High precision ensures accurate identification of attack traffic while minimizing false positives. A high recall reflects the system's ability to detect most attacks, reducing missed threats. Accuracy should be maximized by correctly classifying both attacks and normal traffic. The F1-score, which balances precision and recall, should approach 1, indicating a well-rounded, effective IDS. High values across all these metrics demonstrate a robust and reliable deep learning-based IDS.

Key performance metrics:

- Accuracy (AC): The number of correct forecasts relative to total traffic.

$$AC = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

TP: True Positive: The model correctly predicts the positive class.

TN: True Negative: The model correctly predicts the negative class.

FP: False Positive: The model incorrectly predicts the positive class.

FN: False Negative: The model incorrectly predicts the negative class.

- Precision (P): It is used to figure out how much of the attack traffic is accurately identified.

$$P = \frac{TP}{TP+FP} \quad (2)$$

- Recall (R): It measures the rate of anticipated attacks relative to overall attack traffic.

$$R = \frac{TP}{TP+FN} \quad (3)$$

- F1-score (F1): It provides a more accurate, balanced estimate of both precision and recall.

C. Adaptive DoS Attack Detection - Level 3

Earlier attackers used to flood the network or system with excessive amounts of traffic (ICMP, UDP), data or requests. Modern day attackers do not depend on only one type of DoS Attack. They use timeouts (LDoS), small queries (DNS/NTP Amplification) to bypass the DoS attack detection systems. Effective LDoS attack detection can be achieved using time-based network traffic properties.

Parameters used in the proposed mechanism are listed. Other than $Th_{Discard}$ and $TS_{Duration}$, all values are listed in Table I below:

TABLE I. LIST OF PARAMETERS

Symbol	Description
Con	No. of connections to the server in current Time slot
IA	Packet Inter-arrival time in current Time Slot
AIA	Average Packet Inter-arrival time
TO	No. of Time-outs in current Time Slot
DP	No. of discarded packets in current Time Slot
D	Time Slot Duration (in Seconds)
T	Traffic in current Time Slot
AT	Average Traffic per Time Slot
TH	Threshold number of packets discarded in a particular Time slot

$$AT = \alpha * AT + (1 - \alpha)T \quad (3)$$

$$AIA = \alpha * AIA + (1 - \alpha)IA \quad (4)$$

To analyse network traffic, there are two steps:

Step 1 - The following parameters are calculated using network traffic data from the network interface card: T, TO, DP

Step 2 - Sleeps for D before calculating the following parameters using information gathered from Step 1: IA, AIA, T. It employs the following algorithm using the previously computed parameters. Algorithm 1 details the Adaptive DoS Detection.

Algorithm 1: Adaptive DoS Detection

Step 1: Evaluate T and AT ;

If ($T \geqslant ((1+\beta) * AT)$)

Then Goto Step 2 Else Goto Step 4

Step 2: Evaluate TO with Con and DP with TH;

If ((TO >= 2*Con) and ($DP \geqslant TH$))

Then Goto step 3 Else Goto step 4

Step 3: Evaluate IA with AIA; If ($(AIA \leqslant \gamma * IA)$)

Then Assume Slow Rate DoS attack and not congestion

Else Goto step 4

Step 4: Halt processing till the end of current time slot interval

Traffic and Average traffic are compared. When the volume of traffic in the current time slot T exceeds $(1 + \beta)$ times the volume of that time slot's average traffic (AT), we proceed. We continue when there are more connections than timeouts, and more packets are refused, than the set threshold. The inter-arrival time and its average are then compared. We conclude that we are not experiencing congestion, but rather a Slow DoS assault when IA decreases. The processing is halted for that time period if any of the steps are unsuccessful.

IV. EXPERIMENTAL SETUP AND RESULTS

A. Flow Validation – Level 1

The experimental setup is shown in Fig. 3. The SDN consists of three hosts, namely Host 1 having an IP address of 192.168.0.2, Host 2 having an IP address of 192.168.0.3, and Host 3 having an IP address of 192.168.0.5. Here we use a single controller named Controller 1 with IP address 192.168.0.4. The Attacker having IP Address 192.168.0.1 is an external entity, which does fast, as well as Slow DoS-based attacks, as controlled by us.

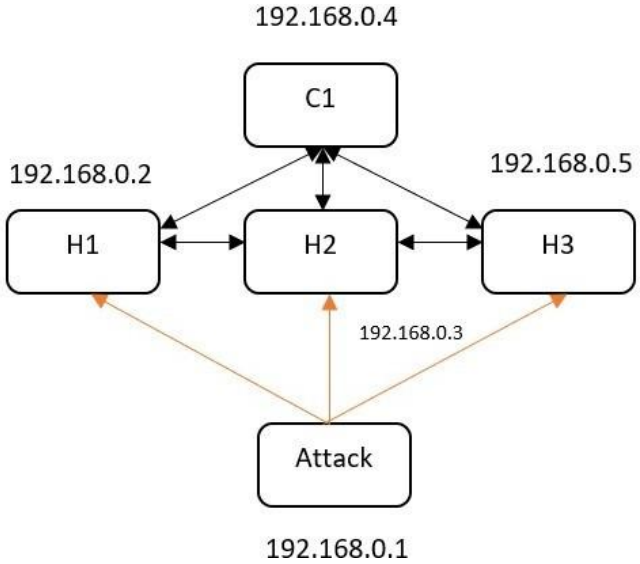


Fig. 3. Experimental setup (level 1).

We have noted their MAC addresses to verify. Data from the switches is extracted in CSV. Here we note the IP address, as well as the MAC Address, which is stored in the external storage. To store this data, remove the colon from the MAC Address and concatenate it with the Source Port Number (which is converted to a string). All the information is stored after applying SHA-256, as shown below. Each entry is stored in Table II to be matched. If the match is found, move to level 2 or wait for approval. During simulation, we found that the frequent IDs are stored and retrieved quickly from the database. New entries are not accepted and need time for approval from the controller. As soon as the entry from the attacker with source port number 6688 arrived, the hash matching failed. The attacker was blocked from moving to level 2, as expected.

TABLE II. CLIENT MATCH TABLE

SrcPort Nr	MAC Address	HashValue	Match
4233	00:0A:95:9D:68:16	b1b05f2632443eff30f3a71e5790298557b27d7d2f969e5859aee7f0081b547e	Yes
5311	00:1B:21:3F:55:A1	1057886baf2908dad4183baf45f653f2591c4db72182a56c6636858361a0b4fe	Yes
9612	00:23:A1:5B:44:65	35f956835f0c51c0aaaf3521f01120ad4a5386f6581031de524cd4f417bb9717	Yes
1239	00:17:CB:01:2A:B3	04674e1d4150b7c464c67277d0c928c6ce4fe52975b5fc32afdb56248a3a1bc6	Yes
6688	00:14:22:01:23:45	2519d99444f45cc88521846451c8b423ec345ecbd96cf069415c88f118c4a29b	No

B. Deep Learning based DoS Detection – Level 2

We have tested three datasets to find the best predictive model so that we can rule out flows based on the performance metrics discussed earlier [48].

The DoS attacks detection results are shown in Table III using the NSL-KDD dataset. The RNN + LSTM model records the highest accuracy of 95.14% in terms of finding DoS attacks. It is also the best in finding precision, recall and achieving F1-score. LSTM and its hybrid models also perform strongly with accuracies of more than 93%. In comparison, the accuracy rate for the SOM model is the lowest at 90.6%. This leads to the observation that, whereas models are useful in DoS detection, advanced deep learning models are on another level in DoS detection.

Table IV illustrates the performance of various machine learning models for detecting DoS attacks using the CIC-DDoS 2019 dataset. The LSTM model achieves the highest accuracy at 95.73%, followed closely by the RNN + LSTM model at 95.60%, showcasing their strong capability in identifying DoS attacks. Precision (P), recall (R), and F1 scores are consistent across all models, with values ranging between 0.85 and 0.88, indicating reliable detection performance across different approaches. However, while accuracy varies slightly, the models demonstrate comparable performance in precision, recall, and F1 score, with LSTM leading in accuracy, but all models performing similarly in overall detection quality. Thus, LSTM proves to be the most accurate, but other models like RNN + LSTM also offer strong results.

TABLE III. NSL-KDD DATASET [45]

Score	*SOM*	*RNN*	*GRU+LSTM+RNN*	*GRU*	*GRU+LSTM*	*LSTM*	*RNN+LSTM*
AC(%)	90.6	92.45	92.95	93.07	93.20	94.34	95.14
P	0.92	0.94	0.94	0.94	0.94	0.93	0.94
R	0.89	0.91	0.92	0.92	0.92	0.91	0.92
F1	0.90	0.92	0.93	0.93	0.93	0.92	0.93

TABLE IV. CIC-DDOS 2019 DATASET [46]

Score	*SOM*	*RNN*	*GRU+LSTM+RNN*	*GRU*	*GRU+LSTM*	*RNN+LSTM*	*LSTM*
AC(%)	90.6	92.53	92.61	93.52	93.67	95.60	95.73
P	0.88	0.88	0.88	0.88	0.87	0.87	0.87
R	0.87	0.87	0.87	0.87	0.87	0.87	0.87
F1	0.86	0.86	0.86	0.85	0.86	0.86	0.86

TABLE V. CIC-IOT 2023 DATASET [47]

Score	*SOM*	*GRU*	*GRU+LSTM+RNN*	*RNN*	*GRU+LSTM*	*LSTM*	*RNN+LSTM*
AC(%)	90.8	92.40	92.81	92.92	93.07	94.23	95.79
P	0.98	1	0.98	0.99	1	0.96	1
R	0.24	0.20	0.24	0.25	0.21	0.40	0.24
F1	0.38	0.33	0.38	0.40	0.35	0.56	0.39

Table V presents the performance of different models for detecting DoS attacks using the CIC-IoT 2023 dataset. The RNN + LSTM model achieves the highest accuracy at 95.79%, followed by LSTM at 94.23%, indicating their effectiveness in identifying DoS attacks in IoT environments. Interestingly, although GRU and GRU + LSTM exhibit perfect precision (P = 1), their recall values are significantly lower, leading to lower F1-scores. This indicates that while these models are highly precise, they miss a substantial portion of true positives. On the other hand, LSTM balances precision and recall, yielding a higher F1-score of 0.56, making it one of the more reliable models in terms of both accuracy and detection performance. While the RNN + LSTM model leads in accuracy, LSTM offers a better overall balance in terms of detection metrics.

The analysis in Fig. 4 highlights the effectiveness of models in accurately detecting DoS attacks on IoT networks, with RNN+LSTM being the most effective. Models like SOM underperform, indicating that not all models are equally suitable for this specific dataset.

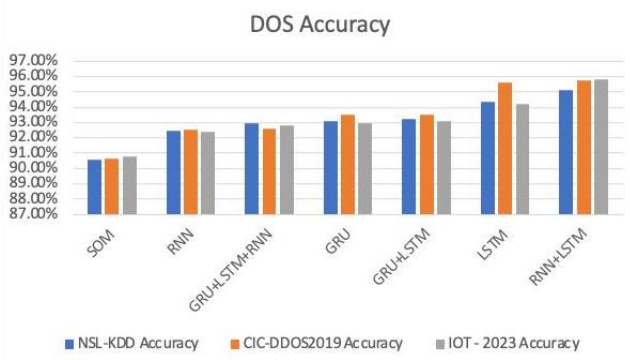


Fig. 4. Comparative Analysis of datasets based on their accuracy for DoS attack.

C. Adaptive DoS Attack Detection - Level 3

The experiment [49] is done using the Mininet Simulator, and we use a Python file to collect the flow, port, and timing statistics. Network simulation is run for 250 minutes, and 1,04,345 rows of data are collected. The simulation is run for a defined interval again, and more data can be collected. The test is divided into two phases. Initially, the simulation is started and kept ready to detect flows. Phase I – Normal simulation is done, allowing the flow of packets across all switches and controllers. Phase II – Attack is simulated, where 100 forged packets are sent. Following values of α, β, γ, and TH are taken as 0.75, 0.30, 0.70 and 1000, respectively. For Phase I: During this phase, data is gathered and data measuring the switch performance is measured, and the data is represented graph. From Fig. 5, it is clear that when there is no attack, the timeouts and discards are null. Also, the packets arrive in an orderly fashion.

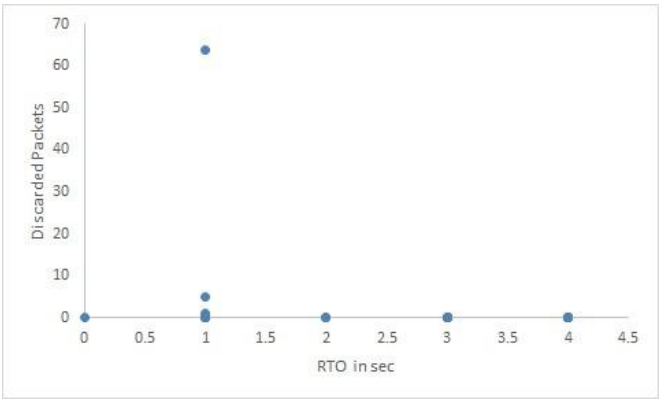


Fig. 5. Normal scenario (RTO vs discarded packets).

The typical traffic exceeds the current traffic in the time slot as soon as the attack begins, as shown in Fig. 6.

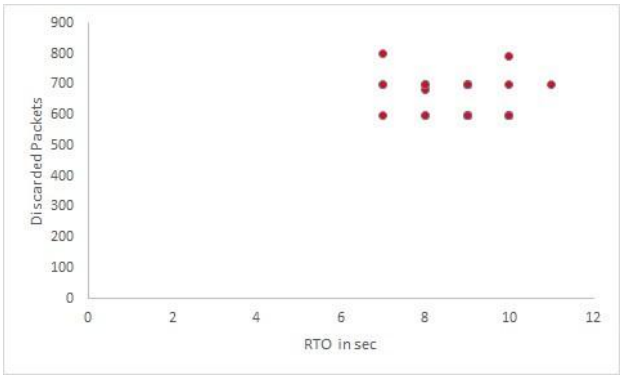


Fig. 6. Attack scenario (RTO vs discarded packets).

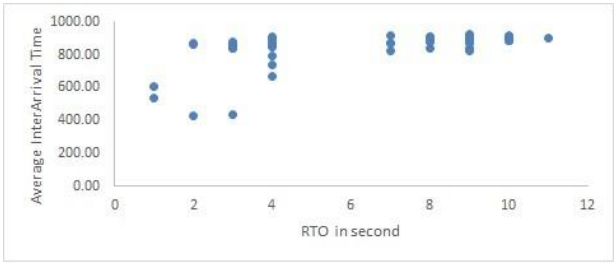


Fig. 7. Interarrival time.

Also, one can observe that the connections are 3 timeouts are 6. The number of packets discarded are way above 600. Looking at Fig. 7, the average interarrival time has gone below the packet interarrival time showing that it could be a low-rate DoS attack rather than congestion.

V. Conclusion

The results clearly indicate that the proposed method is effective against all modern DoS attacks in SDN. The level 1 zero trust approach ensures that only verified clients get access to the SDN. Level 2 based on deep learning, ensures that flows are classified and only verified flows are allowed to move forward. In case when the attacker bypasses levels 1 and 2, the level 3 adaptive DoS detection mechanism will ensure that both fast and low rates are thwarted. Researchers used blockchain for ensuring a zero-trust approach in SDN, which is time-consuming and complex. Hashing ensures that the initial check is done quickly and accurately. Researchers used various hybrid models, but the LSTM+GRU gives accurate predictions with better precision and recall. Researchers working on fast and low-rate attacks haven't worked on a simple, lightweight, combined approach relevant to real-time SDNs. This solution can be integrated with contemporary IDPS, as it is flexible and scalable.

References

[1] Jarraya Y, Madi T, Debbabi M. A survey and a layered taxonomy of software-defined networking. IEEE Commun. Surv. Tutor. 16(4):1955–80,2014.

[2] R. Nagarathna and S. M. Shalinie, SLAMHHA: A supervised learning approach to mitigate host location hijacking attack on SDN controllers, 2017 Fourth International Conference on Signal Processing, Communication and Networking (ICSCN), 2017.

[3] Han T, Jan SRU, Tan Z, Usman M, Jan MA, Khan R, Xu Y. A comprehensive survey of security threats and their mitigation techniques for next-generation SDN controllers. Concurr. Comput.,3–5,2019.

[4] Wang Y, Hu T, Tang G ,Xie J ,Lu J . Sgs: safe-guard scheme for protecting control plane against DDoS attacks in software-defined networking. IEEE Access , 2019.

[5] Shang G, Zhe P, Bin X, Aiqun H, Kui R. FloodDefender: protecting data and control plane resources under SDN-aimed DoS attacks. Proceedings of the IEEE INFOCOM, 2017

[6] Maham Iqbal, Farwa Iqbal, Fatima Mohsin, Dr.Muhammad Rizwan, Dr.Fahad Ah- mad, Security Issues in Software Defined Networking (SDN): Risks, Challenges and Potential Solutions, published in International Journal of Advanced Computer Science and Applications (IJACSA) Vol. 10, No. 10, 2019.

[7] Shang Gao, Zecheng Li, Bin Xiao, and Guiyi Wei,Security Threats in the Data Plane of Software-Defined Networks, published in IEEE Network, pg 108-113.August 2018.

[8] Durner R, Lorenz C, Wiedemann M, Kellerer W,Detecting and mitigating denial of service attacks against the data plane in software defined networks. Proceedings of the IEEE Conference on Network Softwarization: Softwarization Sustaining a Hyper- Connected World: en Route to 5G, NetSoft 2017.

[9] Danping A ,Pourzandi M ,Scott-Hayward S ,Song H ,Winandy M ,Dacheng Z , Threat analysis for the SDN architecture. Tech. Recomm. pp 1-21, July 2016.

[10] Jan J.Laan,Securing the SDN Northbound Interface, Master Research Project sub- mitted at University of Amsterdam. July 2015.

[11] B. Agborubere and E. Sanchez-Velazquez, OpenFlow Communications and TLS Se- curity in Software-Defined Networks, 2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), pp. 560-566, 2017.

[12] Mohamed Mahdy, SDN Southbound Threats, Global Information Assurance Cer- tification by The SANS Institute, 2018.

[13] Thangasamy, Pandikumar, Early Detection of DDoS Attacks in a Multi-Controller Based SDN, published in International Journal of Engineering Science and Computing. 07. 13422-13429, 2017.

[14] Mousavi SM, St-Hilaire M. Early detection of DDoS attacks against SDN con- trollers. In: Proceedings of the International Conference on Computing, Networking and Communications, ICNC 2015; p. 77–81.,2015.

[15] Jiang Y, Zhang X ,Zhou Q ,Cheng Z . An entropy-based DDoS defense mechanism in software defined networks. In: Proceedings of the International Conference on Communications and Networking in China. p. 169–78, Springer; 2016.

[16] Kumar P, Tripathi M, Nehra A, Conti M, Lal C. SAFETY: early detection and mitigation of TCP SYN flood utilizing entropy in SDN. IEEE Trans. Netw. Serv.Manag,15(4):1545–59., 2018

[17] Kalkan K, Altay L, Gu¨r G, Alago¨z F. JESS: joint entropy-based DDoS defense scheme in SDN. IEEE J. Sel. Areas Commun. 36(10):2358–72.,2018

[18] AbdelAzim NM, Fahmy SF, Sobh MA, Eldin AMB. A hybrid entropy-based dos attacks detection system for software defined networks (SDN): a proposed trust mechanism. Egypt. Inform. J.,22(1):85–90, 2021.

[19] Singh J, Behal S. A novel approach for the detection of DDoS attacks in SDN using information theory metric. In: Proceedings of the 8th International Conference on Computing for Sustainable Global Development (INDIACom). IEEE,p. 512–16, 2021.

[20] Mishra A, Gupta N, Gupta B. Defense mechanisms against DDoS attack based on entropy in SDN-cloud using pox controller. Telecommun. Syst.,77(1):47–62, 2021.

[21] AlEroud A, Alsmadi I. Identifying cyber-attacks on software defined networks: an inference-based intrusion detection approach. J. Netw. Comput. Appl.,80:152–64, 2017.

[22] Fan Z, Xiao Y, Nayak A, Tan C. An improved network security situation assessment approach in software defined networks. Peer-to-Peer Netw. Appl.,12(2):295–309, 2019.

[23] Meti N, Narayan D, Baligar V. Detection of distributed denial of service attacks using machine learning algorithms in software defined networks. In: Proceedings of the International Conference on Advances in Computing, Communications and Informatics (ICACCI). IEEE,p. 1366–71,2017.

[24] Makori DO, Ari S.Intelligent hybrid detection and mitigation of distributed denial of service attacks in SDN. Int. J. Eng. Comput. Sci.,6(12):22243–9, 2017.

[25] Myint Oo M, Kamolphiwong S, Kamolphiwong T, Vasupongayya S. Advanced support vector machine-(ASVM-) based detection for distributed denial of service (DDoS) attack on software defined networking (SDN). J. Comput. Netw. Commun. 2019,2019.

[26] Santos, R., Souza, D., Santo, W., Ribeiro, A., Moreno, E., Machine learning algorithms to detect DDoS attacks in SDN. Concurr. Comput. Pract. Exp.,2019.

[27] Xiao Y, Fan Z-j, Nayak A, Tan C-x. Discovery method for distributed denial-of- service attack behavior in SDNS using a feature-pattern graph model. Front. Inf. Technol. Electron. Eng.,20(9):1195–208, 2019.

[28] Tuan NN, Hung PH, Nghia ND, Van Tho N, Phan TV, Thanh NH, A robust TCP- SYN flood mitigation scheme using machine learning based on SDN. In: Proceedings of the International Conference on Information and Communication Technology Con- vergence (ICTC). IEEE,p. 363–8, 2019.

[29] Xu Y, Sun H, Xiang F, Sun Z. Efficient DDoS detection based on k-fknn in software defined networks. IEEE Access,7:160536–45,2019.

[30] Mehr SY, Ramamurthy B. An SVM based DDoS attack detection method for RYU SDN controller. In: Proceedings of the 15th International Conference on Emerging Networking Experiments and Technologies, p. 72–3, 2019.

[31] Dong S, Sarem M. DDoS attack detection method based on improved KNN with the degree of DDoS attack in software defined networks. IEEE Access 2019.

[32] Zhijun W, Qing X, Jingjie W, Meng Y, Liang L. Low-rate DDoS attack detection based on factorization machine in software defined network. IEEE Access 2020,8:17404–18,2020.

[33] Krishnan P, Duttagupta S, Achuthan K. Varman: Multi-plane security framework for software defined networks. Comput. Commun. 2019,148:215–39,2019.

[34] Musumeci F, Ionata V, Paolucci F, Cugini F, Tornatore M. Machine-learning- assisted DDoS attack detection with p4 language. In: Proceedings of the ICC 2020- 2020 IEEE International Conference on Communications (ICC). IEEE,p. 1–6,2020.

[35] Tan L, Pan Y, Wu J, Zhou J, Jiang H, Deng Y. A new framework for DDoS attack detection and defense in SDN environment. IEEE Access,8:161908–19,2020.

[36] Perez-Diaz JA, Valdovinos IA, Choo K-KR, Zhu D. A flexible SDN-based archi- tecture for identifying and mitigating low-rate DDoS attacks using machine learning. IEEE Access 2020,8:155859–72,2020.

[37] Ahuja N, Singal G, Mukhopadhyay D, Kumar N. Automated DDoS attack detection in software defined networking. J. Netw. Comput. Appl.,187:103108,2021.

[38] Tayfour OE, Marsono MN. Collaborative detection and mitigation of DDoS in software-defined networks. J. Supercomput.,1–25, 2021.

[39] Swami R, Dave M, Ranga V. Detection and analysis of TCP-SYN DDoS attack in software-defined networking. Wirel. Pers. Commun.,118(4):2295–317, 2021.

[40] Cui Y, Yan L, Li S, Xing H, Pan W, Zhu J, Zheng X. SD-Anti-DDoS: fast and efficient DDoS defense in software-defined networks. J. Netw. Comput. Appl.,68:65–79, 2016.

[41] Abdulqadder IH, Zou D, Aziz IT, Yuan B. Validating user flows to protect software defined network environments. Secur. Commun. Netw. 2018,2018.

[42] Gao D, Liu Z, Liu Y, Foh CH, Zhi T, Chao H-C . Defending against packet-in messages flooding attack under SDN context. Soft Comput.,22(20):6797–809,2018.

[43] Fichera S, Galluccio L, Grancagnolo SC ,Morabito G ,Palazzo S . Operetta: an openflow-based remedy to mitigate TCP synflood attacks against web servers. Com- put. Netw.,92:89–100,2015.

[44] Mohammadi R, Javidan R, Conti M. Slicots an SDN-based lightweight countermeasure for TCP SYN flooding attacks. IEEE Trans. Netw. Serv. Manag.,14(2):487–97, 2017.

[45] Kaggle, NSL KDD Dataset. 2023. [Online]. Available: https://www.kaggle.com/datasets/hassan06/nslkdd/. [Accessed: Jul. 20, 2024]

[46] UNB, CIC DDoS 2019 Dataset. Canadian Institute for Cybersecurity, University of New Brunswick. [Online]. Available: https://www.unb.ca/cic/datasets/ddos-2019.html. [Accessed: Jul. 20, 2024].

[47] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment," Journal of Sensors, vol. 23, p. 5941, 2023.

[48] H. K. Lamba, H. Gala, R. Mathew, and S. Shinde, "AI based intrusion detection for DDoS and SQL injection attacks," in Proc. 2024 First Int. Conf. for Women in Computing (InCoWoCo), Pune, India, 2024, pp. 1–7.

[49] R. R. Mathew and A. Vidhate, "Adaptive DoS Attack Detection in SDN," 2024 International Conference on Emerging Smart Computing and Informatics (ESCI), Pune, India, 2024, pp. 1-5,

Mid-Upper Arm Circumference Measurement Using Digital Images: A Top-Down Approach with Panoptic Segmentation Using Mask R-CNN

Maya Silvi Lydia[1]*, Pauzi Ibrahim Nainggolan[2], Desilia Selvida[3],
Doli Aulia Hamdalah[4], Dhani Syahputra Bukit[5], Amalia[6], Rahmita Wirza Binti O. K. Rahmat[7]
Department of Computer Science, Universitas Sumatera Utara, Medan, 20155, Indonesia[1, 4, 6]
Computer Vision and Multimedia Laboratory, Universitas Sumatera Utara, Medan, 20155, Indonesia[2, 3]
Department of Public Health, Universitas Sumatera Utara, Medan, 20155, Indonesia[5]
Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, Serdang 43400, Malaysia[7]

***Abstract*—Assessing nutritional status, particularly among children and pregnant women, necessitates accurate measurement of Mid-Upper Arm Circumference (MUAC). This research introduces a novel system for MUAC estimation from digital images using the Mask R-CNN algorithm, employing a top-down panoptic segmentation strategy. The proposed model was designed to identify the upper arm region within human body images and compute MUAC values autonomously. Mask R-CNN was selected due to its capacity to perform precise segmentation of objects within visually complex scenes, especially in the mid-upper arm area. Model training was conducted using a dataset of annotated images, with subsequent evaluation confirming its ability to reliably detect and measure MUAC. The system was validated using 72 image samples, yielding a mean absolute error (MAE) of 2.31 cm when compared to manual measurements. Among these samples, 29.2% (21 individuals) exhibited a measurement discrepancy of 0 to 1 cm, 27.8% (20 individuals) showed a 1 to 2 cm difference, and 43.1% (31 individuals) demonstrated deviations exceeding 2 cm. Despite some variations in measurement accuracy, the system presents a promising tool for enhancing the automation and efficiency of nutritional assessments.**

***Keywords*—Mask R-CNN; mid-upper arm circumference; digital images; segmentation; mean absolute error**

I. Introduction

Recent advancements in artificial intelligence and image processing have significantly impacted the medical and healthcare sectors. A prominent area of application is anthropometry, where these technologies have improved the accuracy and efficiency of assessing nutritional and health status through body measurements [1]. Amongst the various anthropometric indicators, Mid-Upper Arm Circumference (MUAC) has gained recognition as a critical metric for evaluating individual health and physical condition [2] [3] [4].

MUAC is particularly valued for its effectiveness in detecting nutritional deficiencies, especially in vulnerable populations such as pregnant women [5]. As a simple, non-invasive, and rapid screening tool, MUAC is instrumental in identifying individuals at risk of Chronic Energy Deficiency (CED). CED during pregnancy poses a major public health concern, as it is linked to elevated risks of infection, anemia, hemorrhage, and adverse neonatal outcomes, including low birth weight (LBW) [6]. MUAC measurement below the widely accepted threshold of 23.5 cm is generally considered indicative of increased CED risk, necessitating further clinical evaluation and targeted nutritional support [7] [8]. Owing to its low cost, ease of use, and minimal equipment requirements, MUAC is particularly suitable for deployment in limited resource settings [9].

Traditionally, MUAC is measured by wrapping a flexible measuring tape around the midpoint of the upper arm. This process involves identifying anatomical landmarks, specifically the shoulder and the elbow—and determining the midpoint between them. The tape is then positioned horizontally and applied with sufficient tension to avoid measurement errors due to slackness or excessive compression [7]. Despite its practicality, this manual method is susceptible to variability, primarily due to inconsistencies in tape dimensions and user technique, which may compromise the reliability and accuracy of the measurements [10].

The advancement of technology, digital approaches are emerging as potentially more accurate, efficient, and scalable solutions [11][12][13][14][15]. Liu, Sowmya, and Khamis [16] showed that the photogrammetric method can produce precise MUAC measurements, with a Technical Error of Measurement (TEM) value of 3.76 mm and its reliability coefficient (R) reaching 0.99. However, this research has yet to tap the potential of the Mask R-CNN method, which has proven to be highly effective and can be generalized to various tasks, including complex analyses in medical imaging [17][18][19] and human pose estimation [20], making it highly suitable for the detailed segmentation required for accurate MUAC calculation from digital images.

Top-Down Panoptic Segmentation approach utilizing Mask R-CNN is among the pertinent innovations in developing digital image-based MUAC measurement systems. In [20], Mask R-CNN framework represents a robust methodology capable of concurrently performing object detection and generating precise segmentation masks for each detected object instance.

Panoptic segmentation as introduced in [21]. It supports comprehensive scene interpretation by unifying instance and

*Corresponding Author.

semantic segmentation in single framework. This technique assigns each pixel to either distinct objects ("things", such as the upper arm) or homogeneous background regions ("stuff"), thereby generating semantically rich image representations, a critical requirement for anthropometric analysis based on digital images.

The Top-Down approach is particularly advantageous for digital anthropometry, especially when implemented using models such as Mask R-CNN. This method operates in a hierarchical manner, beginning with the detection of regions of interest such as the upper arm through bounding boxes, followed by fine-grained segmentation within these localized areas. Mask R-CNN exemplifies this approach by coupling coarse object detection with subsequent mask refinement. Central to this process is the Region Proposal Network (RPN), which identifies candidate object regions [22] prior to the mask generation stage. This sequential pipeline enhances computational efficiency by concentrating processing resources on the most relevant portions of the image.

Integrating panoptic segmentation with a Top-Down architecture, the accuracy of MUAC measurement is substantially improved. This combination streamlines image analysis workflows, making it a promising innovation for applications in anthropometry and broader health-related fields.

Despite recent progress in digital anthropometry, existing image-based MUAC methods still leave three practical gaps. First, most pipelines do not employ a top-down instance/panoptic segmentation strategy (e.g., Mask R-CNN with RPN/RoIAlign/FPN) to robustly isolate the upper-arm region under real-world variation in pose, clothing, and background, limiting measurement reliability. Second, pixel-to-centimeter scaling is commonly tied to specialized hardware or fixed capture setups, offering no simple, low-cost calibration standardizable across sites in resource-constrained settings. Third, there is limited end-to-end validation against manual MUAC using interpretable metrics that jointly assess region localization (e.g., IoU) and circumference agreement (e.g., MAE in cm) on real subjects. In response, we: i) develop a top-down panoptic-segmentation pipeline based on Mask R-CNN to detect and segment the upper arm from complementary front and side views, ii) formalize an ellipse-based MUAC estimation that fuses both views, and iii) introduce a lightweight, field-feasible pixel-to-centimeter scaling procedure using a ubiquitous wall power outlet as reference. We then evaluate detection/segmentation and MUAC agreement against manual measurements on 72 individuals.

Accordingly, this study develops a top-down panoptic-segmentation pipeline based on Mask R-CNN (with RPN, RoIAlign, and FPN) to detect and segment the upper-arm region from two complementary views (front and side), and formalizes MUAC estimation using an ellipse-based circumference approximation that combines the front- and side-view measurements. To realize field feasibility, the pipeline includes a lightweight pixel-to-centimeter scaling procedure that uses a standardized wall power-plug as a ubiquitous reference object. The system is implemented in Detectron2 [28] and evaluated using Intersection over Union (IoU) [25] for detection/segmentation and mean absolute error (MAE, cm) for MUAC agreement with manual measurements, alongside distributional error analysis and training-configuration comparisons.

The significance of this work is twofold. Practically, it reduces dependence on operator skill and measurement hardware by shifting MUAC assessment to a camera-based, low-cost workflow that can be standardized across sites and is well-suited to limited-resource environments. Technically, it demonstrates how top-down panoptic segmentation can strengthen digital anthropometry by delivering reliable region isolation prior to metric estimation, laying the groundwork for near-real-time and mobile implementations that can support large-scale nutritional screening and follow-up [21].

To guide the reader, the remainder of this study is organized as follows: Section II presents the related works. Section III (Materials and Methods) details the dataset and annotation protocol, image-capture standards, reference-object scaling, the MUAC estimation formulation for front/side views (ellipse-based), the MAE metric, and the Mask R-CNN–based top-down panoptic segmentation pipeline. Section IV (Implementation and Results) first describes the dataset preparation and training environment, then details the system architecture. It also presents a comparative evaluation across training iterations and model variants (loss and accuracy), and reports the test results—detection IoU; MUAC–vs–manual agreement summarized by MAE (in cm) with its error distribution; and qualitative top-down panoptic-segmentation visualizations. Section V (Discussion) analyzes the findings, limitations, and avenues for improvement. Section VI (Conclusion) summarizes the contributions and key findings. Finally, acknowledgments are provided.

II. Related Work

Prior digital anthropometry solutions span: i) manual tape measurement, ii) monocular photogrammetry and regression, iii) 3-D imaging systems, and iv) learning-based detection/segmentation pipelines.

1) Manual tape. MUAC is commonly measured by wrapping a dedicated MUAC tape around the upper arm; however, design differences (e.g., material thickness and scale placement) introduce systematic measurement discrepancies, and there is no single global tape specification—prompting calls for common design standards and standardized reporting [10].

2) Single-camera photogrammetry/regression. Liu et al. showed that MUAC (and height) can be estimated from images using linear regression under controlled capture with high reliability (TEM$\approx$3.76 mm, R$\approx$0.99). These systems typically rely on strict imaging geometry and explicit calibration/landmarking, and do not perform instance-level region isolation of the upper arm [16].

3) 3-D imaging. Automated 3-D anthropometry has demonstrated precise body measurement extraction, including in low-resource settings, but requires dedicated hardware and controlled environments that can increase deployment cost and operational complexity [11], [29], [14].

4) Learning-based segmentation. Mask R-CNN and related top-down components (RPN, RoIAlign, FPN) [20], [24] deliver strong instance segmentation across diverse medical-imaging tasks, including organ and lesion analysis [17]–[19]. To our knowledge, integrating this paradigm into an end-to-end MUAC pipeline—with explicit centimeter scaling and paired-view fusion—remains underexplored.

Our study advances category (iv) by: 1) using a top-down Mask R-CNN pipeline to robustly isolate the upper-arm region from both front and side views, 2) fusing the views via an ellipse-based circumference formulation, and 3) introducing a lightweight, field-feasible pixel-to-centimeter calibration using a ubiquitous wall power outlet as the reference object. We evaluate both localization (IoU) and agreement with manual MUAC (MAE), on 72 subjects, thereby reporting interpretable, end-to-end accuracy rather than proxy metrics alone. Compared to (ii), our approach avoids hand-crafted landmarking and yields explainable, mask-level measurements; compared to (iii), it requires only commodity cameras and a common reference object, improving deployability in resource-constrained settings [16], [11], [29].

Across categories, open challenges remain: a) robust segmentation under clothing–background similarity and sub-optimal pose; b) calibration methods that are reference-free or self-calibrating; c) generalization beyond standardized capture; and d) near-real-time, mobile deployment. These gaps motivate methods that combine reliable instance-level isolation with simple, field-feasible calibration and interpretable circumference estimation.

III. Materials and Methods

This section establishes the methodological foundation of the proposed MUAC-from-image system. We first outline the Mask R-CNN–based top-down panoptic segmentation pipeline—covering the backbone-FPN architecture, RPN/ROIAlign heads, and instance masks—then formalize MUAC estimation from front/side views via an ellipse-based circumference formulation. We also define the MAE metric and detail the dataset acquisition and image standards, COCO-formatted annotations (person, hand, upper arm) prepared in Roboflow, and the pixel-to-centimeter scaling derived from a wall power-plug reference.

A. Mask R-CNN

In the realm of instance segmentation, Mask R-CNN introduces a substantial enhancement, enhancing the Faster R-CNN framework by integrating both object detection and detailed pixel-wise delineation of individual objects [20]. Its primary innovation lies in the addition of a separate, parallel pathway dedicated to producing binary segmentation masks for each proposed region, known as a Region of Interest (RoI). This structure allows the model to simultaneously perform object classification, precise localization, and segmentation at the instance level.

The model employs a two-stage processing pipeline, as illustrated in Fig. 1. Initially, the Region Proposal Network (RPN) [22] identifies candidate object regions within the input image, generating a set of preliminary bounding boxes. These proposals are subsequently refined in the second stage, where features corresponding to each RoI are extracted using the RoIAlign operation. This operation is critical for maintaining spatial alignment between the input feature map and the sampled features, which directly contributes to the precision of the segmentation output. Utilizing these aligned features, the network proceeds to classify each object, adjust the bounding box coordinates, and produce a binary segmentation mask delineating the object's spatial boundaries.

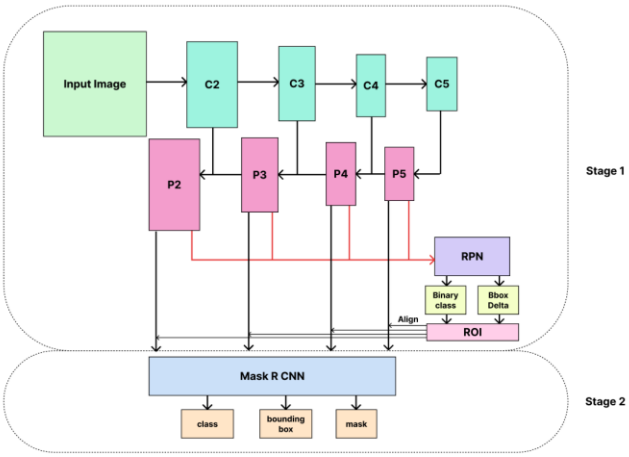


Fig. 1. Architecture of Mask R-CNN.

A key advancement in Mask R-CNN is the replacement of the RoIPool layer with RoIAlign. RoIPool's quantization step often introduces spatial misalignments that compromise mask accuracy. RoIAlign mitigates this issue by avoiding quantization, preserving the exact spatial information during feature extraction. This refinement substantially improves both detection precision and the quality of the generated masks.

Since its introduction, Mask R-CNN has consistently demonstrated cutting-edge performance across prominent visual recognition benchmarks, particularly the COCO (Common Objects in Context) dataset [23]. Its exceptional accuracy has contributed to its broad adoption in diverse computer vision applications, extending well beyond its original roles in instance segmentation and object detection to encompass more advanced tasks, including human pose estimation.

The operational workflow of the Mask R-CNN architecture is organized into two main phases. Initially, a deep convolutional neural network—typically a ResNet variant—serves as the backbone to derive hierarchical feature maps from the input image. The extracted features are then passed to the RPN, which identifies potential object-containing regions. In the subsequent phase, the RoIAlign mechanism is employed to extract accurately aligned features corresponding to each proposed region. These refined features are subsequently utilized to simultaneously carry out object categorization, refine bounding box coordinates, and generate segmentation masks at the pixel level for individual instances.

In first stage, backbone network typically ResNet produces multi-scale feature maps denoted as C2 through C5, corresponding to different levels of convolutional hierarchy. These maps capture progressively abstract representations, with

deeper layers encoding semantic information and shallower layers retaining fine-grained spatial details.

To address scale variation among objects, Mask R-CNN integrates a Feature Pyramid Network (FPN) [24]. FPN constructs a multi–level feature hierarchy (P2–P5) by combining high-level semantic features from deeper layers with spatially detailed information from earlier layers through a top–down pathway and lateral connections. This multi–scale representation enhances the model`s ability to accurately detect and segment objects of various sizes within the same image.

Once the feature map is formed, RPN generates proposals for areas potentially containing objects. Each area proposal is assessed based on two main outputs:

- Binary Class: Determines whether the area contains objects or not.
- Bounding Box Delta: Correction of the bounding box on the proposal area to improve the accuracy of object position prediction.

Next, the area proposals generated by RPN are processed using ROI Align, a technique that aims to extract features with a fixed size, even though the size of the area proposals varies. ROI Align improves the shortcomings of the ROI Pooling method, which tends to produce quantization errors in the detection process.

During the second phase, the region proposals produced by the Region Proposal Network (RPN) are further processed by the Mask R-CNN head, which simultaneously carries out object classification, refines bounding box coordinates, and generates segmentation masks. This stage produces three principal outputs:

- Object Classification: The model assigns a category label to each detected object based on its learned features.
- Bounding Box Regression: The system adjusts the coordinates and dimensions of the bounding boxes to more accurately enclose the detected objects.
- Mask Prediction: In addition to detecting objects, Mask R–CNN produces a binary mask for each identified instance. This mask delineates the specific pixels associated with the object, enabling precise instance-level segmentation within the image.

This stage enables Mask R–CNN to provide outputs like bounding boxes, class labels, and binary masks that map object areas more accurately. Thus, Mask R–CNN is not only able to detect the position of objects but also provide detailed segmentation for each detected object. Final result of Mask R–CNN visualized by displaying the bounding box, class label, and mask on the input image.

B. Measurement Upper Arm Circumference (MUAC)

MUAC serves as a simple yet effective anthropometric measure for evaluating the nutritional status of pregnant women, particularly in screening for CED. Measure value below 23.5 cm is recognized as indicative of elevated CED risk, which is associated with adverse maternal outcomes and an increased likelihood of delivering infants with LBW.

Traditionally way assessed using manual measuring tapes, MUAC in this study is estimated through a digital method that leverages upper arm image segmentation. The approach begins with Mask R-CNN, which detects and accurately segments the upper arm within the image. From the segmentation mask's bounding box, specific geometric features—namely diagonal and horizontal lengths—are extracted to estimate MUAC.

For images captured from a frontal view, MUAC is inferred from the length of a diagonal line extending from the top-right to the bottom-left corner of the bounding box. In lateral view images, estimation is based on the horizontal distance from the midpoint of the left vertical edge of the bounding box to its geometric center. This view-dependent measurement technique is adapted from established methodologies [16], which also employed diagonal and horizontal representations to approximate upper arm dimensions. The proposed automated method aims to enhance the precision and consistency of MUAC estimation in digital anthropometric analysis.

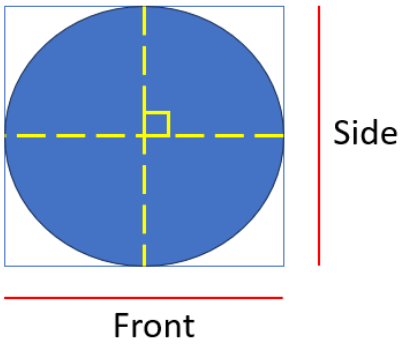


Fig. 2. A method for measuring MUAC.

The calculation method for MUAC is shown in Fig. 2. The front is the result of MUAC measurement with the picture taken from the front, and the side is the result of MUAC measurement with the picture taken from the side. Because the shape of the MUAC is closer to an ellipse than a circle, we will use the elliptical equation [see Eq. (1)]:

$$\text{Circumference} = 1/2\pi\ (\text{Front} + \text{Side}) \quad (1)$$

C. Mean Absolute Error (MAE)

MAE is one of the most frequently used evaluation metrics to measure the error rate in a regression or forecasting model. MAE quantifies the average magnitude of the absolute differences between observed values and corresponding model predictions, providing a straightforward indication of the model's predictive accuracy [26].

MAE is a commonly used performance metric that quantifies the average magnitude of errors between predicted values ($\hat{y}_i$) and actual ground truth values (y_i). It calculates the absolute differences without taking into account the direction of the errors, thereby providing a clear and intuitive interpretation in the same units as the original data. Due to this property, MAE is particularly well-suited for image-based measurement applications, such as those investigated in this study. The mathematical formulation of MAE is presented as follows [see Eq. (2)]:

$$MAE = \frac{1}{n} i + \sum_{i=1}^{n} (yi - \hat{y}i) \quad (2)$$

where,

- yi is the manual measurement value (reference)
- $\hat{y}i$ is the system measurement value
- n is the amount of data

D. Dataset

This study collected the dataset in person at the Faculty of Public Health, University of North Sumatra. Pictures were taken while the individual stood upright, with the left hand unobstructed by clothing and raised slightly, while the right hand was pointed straight down, as shown in Fig. 3. There was a total of 392 images, as shown in Fig. 4, but only 144 images were suitable for MUAC measurement. The reason is three-fold: i) Four pictures were taken from different angles during data collection. Only two images per individual were used in this study, so the remaining two were discarded. ii) The profile data of the individual in the image was not found, making the image unusable. iii) The captured image does not conform to the image standard.



Fig. 3. A dual-angle picture in-pose for MUAC measurement.



Fig. 4. The annotated set of images.

Images that do not meet the image standard cannot be included in this study because testing results will differ significantly from those that meet the image standard. Also, the measurement results will not match because the images that do not meet the image standards have different camera-to-object distances. The image standard detail such as Dimension: 2340 × 4160 pixel; Resolution:72 DPI; File Type: JPG; Camera to object distance: 180 cm. The collected data set was manually annotated on the Roboflow platform [27]. Annotation is done by marking bounding areas and bounding boxes to the objects to be detected in the image. The images in this dataset are put into three classes: person, hand, and upper arm. One hundred forty-four images were manually annotated. Each person has two images: a front-view image and a side-view image. The data labeling is done on a total of 72 individuals.

After annotating the images, a standard dataset suitable for model training was built. The dataset creation process was performed using the same annotation platform. This dataset is generated in JSON format, which is adapted to the COCO (Common Objects in Context) format so that it can be used directly in the Mask R-CNN architecture.

We used an in-house, task-specific dataset—rather than public corpora—because generic scene datasets lack MUAC-specific labels, paired front/side views, and the controlled capture conditions described above that are required for reliable pixel-to-centimeter scaling. These properties align with our method's needs (upper-arm detection/segmentation and dual-view geometry) and enable a faithful end-to-end evaluation with Mask R-CNN. We acknowledge the modest dataset size and controlled setting as limitations and plan to expand to more diverse populations and capture conditions in future work.

E. Reference Object Method

To establish a scale for measurements within the image, we employ a reference object method by using a known object—a standard power plug- in this study—as a reference for scaling. This approach relies on the known dimensions of the reference object to infer measurements within the image. The power plug is a standard object because its dimensions are typically standardized and can be reliably measured.

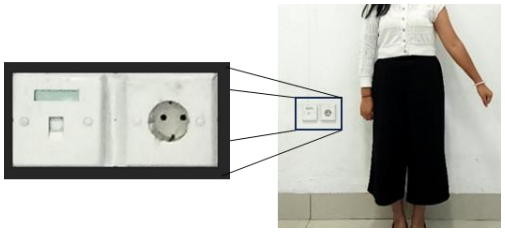


Fig. 5. (Left) Locating the power plug on the left side of the wall, and (Right) cropping the power plug with an image editor.

In Fig. 5, two images are presented to illustrate the process. The Right image shows an MUAC individual standing in front of a wall with a power plug visibly mounted on it. This initial image captures the scene as it appears naturally, with the power plug as the calibration reference object. The Left image in Fig. 5 shows the result after the power plug has been isolated by cropping the original image using Adobe Photoshop. This cropping step focuses on the power plug, removing extraneous elements from the image to allow for precise measurement.



Fig. 6. Determining the power plug's height using a measuring tape.

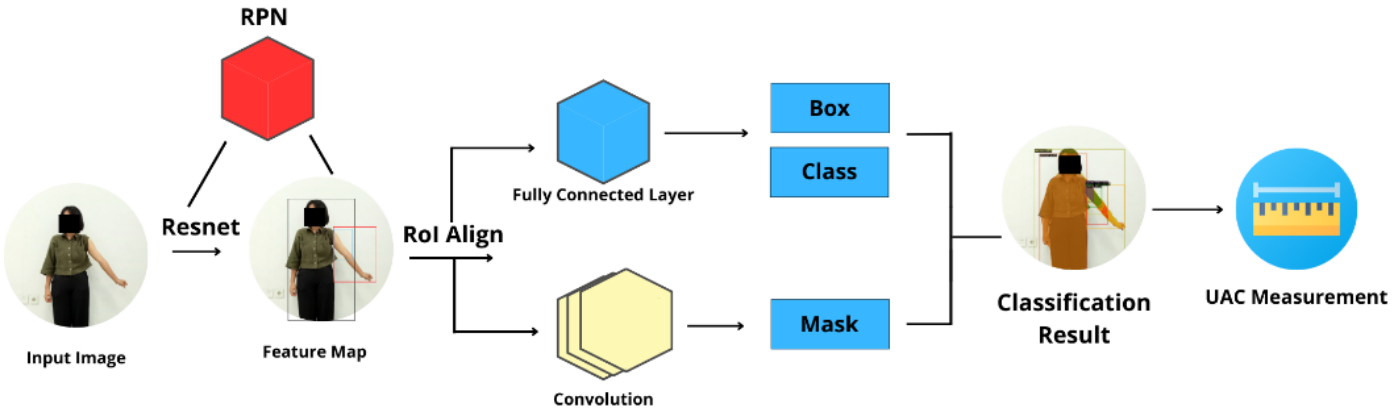


Fig. 7. General architecture of proposed system.

Fig. 6 shows the physical measurement of the actual power plug using a measuring tape, which determines its height to be 8.5 cm. Together, the equivalent height in pixels after measuring the power plug with digital imaging software is the width 297 pixels and height 136 pixels. These measurements are critical for calculating the scale factor that relates pixel dimensions in the image to real world dimensions.

The scale is calculated using the following equation:

$$Scale = y : \hat{y} \tag{3}$$

The variable y refers to the physical height of the power plug (8.5 cm, and the variable $\hat{y}$ refers to the height of the power plug in pixels (136 pixels). After plugging the values measured beforehand into the equation, the scale is computed: Scale = 8.5 cm: 136 px.

Simplifying this ratio, we divide both sides by 8.5, yielding 1 : 16. This means that 16 pixels in the image correspond to 1 cm in the real world. In other words, the pixel-to-centimeter conversion factor is 16 pixels/cm [see Eq. (3)]. This scale factor is then applied to all object detection and segmentation results obtained from the image. For example, if an object detected in the image has a height of 48 pixels, its real-world height would be 48 / 16 = 3 cm.

IV. Implementation and Results

This section presents the implementation and empirical findings: dataset preparation and training on Google Colab using Detectron2 with Mask R-CNN-FPN, the end-to-end system architecture, and a performance study across training iterations (1000 versus 2000). We then report testing and evaluation results—including IoU-based detection accuracy, MUAC measurement comparisons with the resulting MAE, and the accuracy distribution—and illustrate the top-down panoptic segmentation outputs that underpin the measurements.

A. Dataset and Training

In this study, the dataset was prepared and processed using the Roboflow platform, a tool specifically developed to optimize workflows in computer vision applications. During the data collection phase, relevant images were gathered and subsequently organized within Roboflow to ensure compatibility with model training requirements. This preparation process likely encompassed several key steps, including image annotation (e.g., labeling objects within images), data augmentation (e.g., applying transformations such as rotation or scaling to enhance dataset variability), and partitioning the dataset into training, validation, and test subsets. Roboflow's integrated functionalities—such as automated preprocessing and annotation tools—facilitate the creation of well-structured and machine learning-ready datasets.

Following the completion of dataset preparation, model training was conducted using Google Colab (Google LLC). Cloud-based platform offers access to significant computational resources, including Graphics Processing Units (GPUs), at little to no cost. Google Colab was using as the training environment primarily due to its robust GPU support, which is essential for efficiently accelerating the computationally intensive process of training deep learning models. The training efforts within this environment centered explicitly on the Mask R-CNN model, a novel architecture renowned for its efficacy in object detection and instance segmentation.

This research employs Detectron2 [28], a comprehensive library developed by Facebook AI Research (FAIR). Detectron2 facilitates the execution of complex, high-precision inference tasks, a capability leveraged in this study for model evaluation on the prepared dataset. For the specific architectural implementation herein, the Detectron2 framework was configured to utilize the Mask R-CNN model in conjunction with FPN.

B. System Architecture

The diagram depicts the data flow through the system, starting from the input images, followed by feature extraction using a backbone network. The system then branches into two main tasks: 1) a box classification and regression head, which predicts the object class and refines the bounding box coordinates, and 2) a mask prediction head, which generates a segmentation mask for each detected object. The final output is designed to produce measurements for the MUAC using the aforementioned algorithms. Fig. 7 illustrates the general architecture of the proposed system, providing a visual overview of the workflow.

C. Model Performance Evaluation with Varying Iterations

To evaluate the impact of training iterations on model performance, we compare the total loss and classification loss of different models trained for 1000 and 2000 iterations, as shown

in Table I. Total loss is a comprehensive metric that measures the overall error of the model, encompassing components such as classification, bounding box regression, segmentation, and object proposal. Classification loss quantifies the model's error in assigning objects to their correct classes. Lower values for both metrics indicate improved model performance.

TABLE I. PERFORMANCE COMPARISON FOR ALL TESTED MODELS

Model	Accuracy (P)	Total Loss	Class Loss
Faster R–CNN R50-FPN-3x (1000 iterations)	0.96	0.33	0.08
Faster R–CNN R50-FPN-3x (2000 iterations)	0.98	0.19	0.05
Mask R–CNN R50-FPN-3x (1000 iterations)	0.96	0.41	0.08
Mask R–CNN R50-FPN-3x (2000 iterations)	0.98	0.26	0.04
Mask R–CNN R101-C4 (1000 iterations)	0.98	0.40	0.04
Mask R–CNN R101-C4 (2000 iterations)	0.99	0.26	0.02

Table I provides a comparative summary of performance results for multiple models, including Faster R–CNN and Mask R–CNN variants. For instance, as illustrated by data also visualized in Fig. 8, extending the training of the Mask R–CNN R50-FPN-3x model from 1000 to 2000 iterations led to a reduction in total loss from 0.41 to 0.26 and in classification loss from 0.08 to 0.04. Similarly, for the Faster R-CNN R50-FPN-3x model, an identical increase in training iterations (from 1000 to 2000) yielded a decrease in total loss from 0.33 to 0.19 and classification loss from 0.08 to 0.05. This trend of improved loss metrics with more extensive training was consistent across other evaluated models. For example, the Mask R-CNN R101-C4 model exhibited a drop in total loss from 0.40 to 0.26 and in classification.

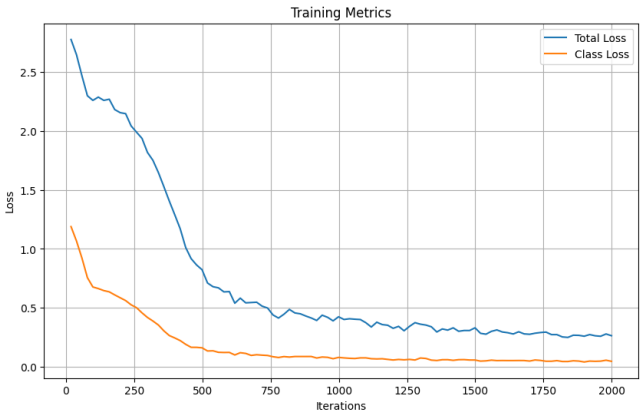


Fig. 8. Training loss curves for Mask R-CNN R50-FPN-3x. The plot illustrates the decrease in total loss and class loss over 2000 training iterations. The continued reduction in both loss metrics suggests that training for 2000 iterations yields a more refined model compared to training for only 1000 iterations.

Consistent with the observed improvement in loss metrics, the accuracy (P)—the proportion of correct predictions—also increased with more training iterations. For example, the accuracy of the Mask R-CNN R50-FPN-3x model improved from 0.96 to 0.98 when its training was extended from 1000 to 2000 iterations. Similarly, the Mask R-CNN R101-C4 model's accuracy rose from 0.98 to 0.99 under the same increase in training duration.

The presented results demonstrate that increasing training iterations from 1000 to 2000 consistently reduces total and classification losses across all tested models. This reduction is indicative of enhanced model convergence and lower overall error. The concurrent improvements observed in accuracy further corroborate this finding. Therefore, predicated on these empirical observations, this study adopts models trained for 2000 iterations, as they exhibit demonstrably superior performance characterized by minimized error and heightened accuracy.

D. Testing and Evaluation

Following the completion of model training, evaluation and testing procedures are conducted within the Colab notebook environment, where inference results are visualized directly on the input images. This workflow is seamlessly integrated with cloud storage services, which are utilized to manage both the datasets and the trained model files. To assess the accuracy of object detection, the IoU metric is employed as a key performance indicator. The IoU results for all evaluated images are presented in Table II.

TABLE II. IOU CALCULATION RESULTS

Individual	IoU for front-view image	IoU for side-view image
1	0,78	0,75
2	0,80	0,82
3	0,79	0,84
4	0,90	0,73
5	0,87	0,83
6	0,77	0,61
7	0,90	0,78
8	0,87	0,79
9	0,82	0,87
10	0,83	0,79
11	0,81	0,78
12	0,81	0,83
13	0,0	0,63
14	0,81	0,76
15	0,82	0,70
16	0,86	0,92
17	0,81	0,74
18	0,85	0,90
19	0,88	0,87
20	0,83	0,71
21	0,89	0,69
22	0,80	0,62
23	0,83	0,82
24	0,81	0,70
25	0,85	0,85
26	0,78	0,78

27	0,88	0,63
28	0,83	0,69
29	0,85	0,88
30	0,76	0,79
31	0,81	0,78
32	0,87	0,67
33	0,89	0,79
34	0,73	0,64
35	0,78	0,85
36	0,89	0,68
37	0,76	0,81
38	0,89	0,82
39	0,89	0,88
40	0,79	0,88
41	0,85	0,70
42	0,72	0,59
43	0,76	0,79
44	0,81	0,88
45	0,86	0,70
46	0,76	0,69
47	0,87	0,76
48	0,85	0,69
49	0,76	0,79
50	0,89	0,77
51	0,79	0,65
52	0,83	0,79
53	0,89	0,68
54	0,80	0,68
55	0,82	0,88
56	0,79	0,78
57	0,86	0,88
58	0,89	0,85
59	0,87	0,74
60	0,85	0,84
61	0,81	0,79
62	0,86	0,88
63	0,91	0,73
64	0,88	0,81
65	0,82	0,71
66	0,74	0,76
67	0,87	0,69
68	0,87	0,86
69	0,90	0,82
70	0,81	0,77
71	0,85	0,78
72	0,83	0,80

After inference, the output is extracted to the CPU for easier processing. From here, the bounding boxes, masks, class labels, and prediction scores are extracted from the results. The class indices corresponding to the objects to be detected are then identified, namely "person", "hand", and "upper arm".

Next, iterations are performed on each class to find the best detection based on the highest score value of each detected object. If the processed class is "upper arm", then the mask's diagonal and horizontal lengths are calculated using the CalculateDiagonalLength and HorizontalCutLength algorithms. The best mask for each class is then saved for further visualization. The saved information includes the bounding box, mask, score, and class label of the detection result. Subsequently, after measuring the MUAC of 72 individuals, the results obtained can be seen in Table III.

TABLE III. COMPARISON OF MUAC MEASUREMENTS

Individual	Manual measurement (cm)	System measurement (cm)
1	22,5	25,61
2	26	25,71
3	29	31,69
4	27,5	32,58
5	24	24,92
6	32,5	29,73
7	29	31,60
8	25,5	26,20
9	23,5	24,73
10	27	31,20
11	32	33,17
12	29	29,63
13	28,5	54,46
14	24	25,32
15	26	24,33
16	26,5	26,98
17	24	26,39
18	26	29,04
19	21,5	26,98
20	24	26,69
21	23	26,49
22	27	25,12
23	28	32,09
24	29	28,65
25	33	33,26
26	24	24,53
27	28	30,81
28	27	29,93
29	22	23,84
30	27	29,53
31	28	31,50
32	21	23,94
33	30,5	31,30
34	26	26,59
35	26	27,18

36	35	40,52
37	35	32,09
38	26,5	29,14
39	27,5	28,95
40	25,5	26,39
41	23	23,25
42	27	25,32
43	27	30,52
44	22	23,16
45	29	33,36
46	27	25,81
47	26,5	28,36
48	24,5	23,35
49	29	32,58
50	26	30,12
51	30	35,62
52	20	20,02
53	23	23,16
54	23	24,92
55	25,5	26,30
56	23	23,65
57	29	32,97
58	29	26,79
59	22	20,80
60	30	29,53
61	24	25,90
62	29,5	29,63
63	24	25,51
64	31	31,4
65	30	32,18
66	32	34,05
67	29	29,14
68	31,5	33,85
69	28	29,24
70	26	26,79
71	20,5	21,88
72	26	27,08
73	22,5	25,61
74	26	25,71

The overall system accuracy can be seen in Table IV, which provides a breakdown of the system's accuracy when tested on a dataset of 72 individuals using Google Colab as the training and inference platform.

TABLE IV. ACCURACY DISTRIBUTION

Total Data	72 Individuals
0 - 1 cm Results	21 Individuals
1 - 2 cm Results	20 Individuals
Results > 2 cm	31 Individuals

The distribution of results presented in Table IV elucidates the system's overall accuracy profile and identifies key limitations. While a substantial proportion of MUAC predictions achieve high accuracy (29.2% exhibiting a 0–1 cm difference from actual values) and moderate precision (27.8% with a 1–2 cm difference), the considerable percentage (43.1%) of predictions deviating by more than 2 cm suggests a lack of uniform precision across the study population. This inconsistency in performance could plausibly arise from various factors. These may include disparities in lighting conditions during image acquisition, instances of occlusion affecting the arm region, or natural variations in the anatomical appearance of the MUAC region across individuals, all of which can adversely affect the performance of the underlying Mask R–CNN model responsible for object detection and segmentation.

An evaluation was conducted using the Mean Absolute Error (MAE) metric, as defined in Eq. (2), to assess system performance quantitatively. This metric computes the mean absolute deviations of the system-generated measurements and the manual (ground truth) measurements for all individuals in the dataset. The analysis yielded an MAE of 2.31 cm. This result, representing the average discrepancy between the system's estimates and the actual values, highlights the necessity for additional improvements to improve the system's measurement precision.

E. Top-Down Panoptic Segmentation Approach

The system's measurement process was visualized using Detectron2's Visualizer, a dedicated utility for illustrating the outcomes of object detection and segmentation procedures. This tool graphically depicts the boundaries and masks of detected objects (specifically the upper arm in this context), superimposing key measurement indicators, such as diagonal and horizontal lines, onto the segmented areas. Such visual outputs are invaluable for qualitatively evaluating the system's detection and segmentation performance. Furthermore, they play a critical role in validating quantitative measurements by enabling direct comparison with ground truth data, which ultimately supports the assessment of the system's output reliability.

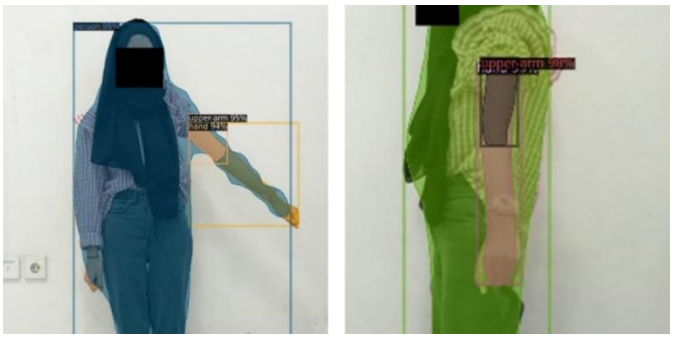


Fig. 9. Top-down detection and segmentation results for individual 52.

Fig. 9 is an example case with an estimated MUAC measurement that is the closest to the ground truth. The length of the diagonal line intersecting the detected mask area is 84 pixels. Meanwhile, the horizontal line bisecting the mask is 120 pixels long. The measurement results in pixel units are then

converted into centimeter (cm) units using a conversion factor, where 1 cm is equivalent to 16 pixels [derived from Eq. (3)].

After the conversion, it is found that the length of the diagonal line intersecting the mask (from the front image) is 5.25 cm, while the length of the horizontal line intersecting the mask (from the side image) is 7.5 cm. Next, these values are used in, as shown in Eq. (1), the elliptical formula to calculate the overall MUAC, which is 20.0175 cm.

For further examples, the following two instances are system MUAC measurements with a significant difference from the corresponding ground truth MUAC.

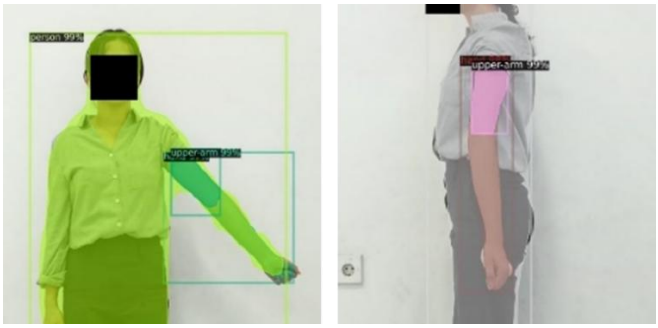


Fig. 10. Top-down detection and segmentation results for individual 48.

Fig. 10 illustrates an example of an image showing the result of a MUAC measurement that deviates by 1 to 2 cm from the ground truth value. In the image, the length of the diagonal line intersecting the detected mask area is 98 pixels (converted to 6.125 cm). Meanwhile, the length of the horizontal line bisecting the mask is 140 pixels (converts to 8.75 cm). Based on the data presented in Table III, the ground truth of individual 48 is 24.5 cm. Thus, the difference between the computed result and the ground truth is 1.15 cm.

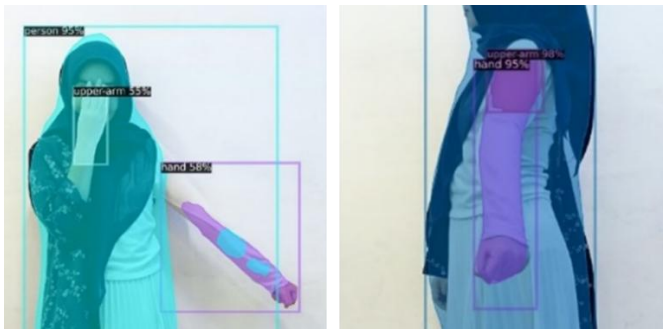


Fig. 11. Top-down detection and segmentation results for individual 13.

In another case, Fig. 11 is an example of images with the results of a MUAC measurement with a difference of > 2 cm compared to the actual MUAC measurement. The diagonal line intersecting the mask is 346 pixels (converts to 21.65 cm), while the horizontal line intersecting the mask is 209 pixels (converts to 13.06 cm). Plugging these values into the elliptical formula, we get the circumference measurement of 54.46 cm.

V. Discussion

The results of the IoU calculation in Table I shows that the model achieves reasonably good accuracy. The average for IoU calculation results for images taken from the front reaches 0.82, which is quite close to 1. The average IoU calculation results for images taken from the side reach 0.77, shows that the model is better at detecting pictures taken from the front than those taken from the side.

In Table III, the actual MUAC of individual 52 is 20 cm. So, the calculation result in this case has a difference of about 0.15 cm with the ground truth. The difference is not too high due to the person doing a moderately good pose where the arm is raised high enough so that the system can detect the upper arm well. The person also raises their sleeves, so that the upper arm is more straightforward to detect.

In some cases, as seen with individual 13, there is a notable variation between the estimated MUAC and the ground truth. In the case of individual 13, the size difference reaches 25.96 cm. This size difference is important because the system does not detect the upper arm correctly. As can be seen in Fig. 11, the system detects the upper arm on the person's left hand. This detection error is caused by the individual wearing a white shirt that resembles the background color of the picture. Furthermore, the participant's sleeve remained unraised, a circumstance that posed an additional challenge to the system's ability to perform accurate detection and segmentation of the target region.

Future Work: Building on our error analysis—evidenced by the front- versus side-view IoU gap (0.82 vs. 0.77) and failures such as individual 13, where clothing–background similarity and unraised sleeves led to mis-segmentation—we outline four concrete directions with measurable targets: 1) Robustness to clothing – background similarity and atypical poses: targeted augmentations (color/texture jitter, occlusion/cutout, pose perturbation), hard-negative mining, and boundary-refinement heads; target $\geqslant$ +0.03 absolute IoU on side view and $\geqslant$ 20% reduction in MUAC errors > 2 cm. 2) Calibration beyond site-specific objects: ablate the current outlet-based scaling against portable fiducials (checkerboard/ArUco) and reference-free/self-calibrating options; target $\leqslant$ 1.5$\times$ variance relative to the best method and < 5% catastrophic scale failures. 3) View fusion and measurement modeling: benchmark the current ellipse formulation against (a) robust statistics over multiple per-view cuts and (b) a small interpretable regressor from per-view mask features; target $\geqslant$ 0.3 cm MAE reduction without sacrificing interpretability. 4) Real-time, on-device deployment and prospective validation: compress the model (pruning/quantization), profile on mobile hardware, and run a head-to-head study against trained operators using standardized MUAC tapes; target $\leqslant$ 200 ms end-to-end latency per view with $\leqslant$ +0.2 cm MAE degradation versus desktop, intraclass correlation $\geqslant$ 0.9, and Bland – Altman limits of agreement within $\pm$3 cm [29].

VI. Conclusion

The Mask R-CNN model accurately detects the upper arm and other body parts. This shows good potential for object segmentation, especially in the case of MUAC measurement. Based on the IoU calculation, 0.82 was obtained for images taken from the front angle and 0.77 for pictures taken from the side. These high IoU values indicate that the applied Mask R-CNN model can produce segmentations close to the annotated ground truth mask, especially in images that meet the image

standards. This value also indicates that the model can detect upper-arm objects better when the image is taken from the front. This result supports the effectiveness of the system for MUAC measurement.

The MUAC measurement results obtained through the digital imaging system show variations in accuracy depending on the quality of the image. As seen in Fig. 9, when the photos are taken with closer compliance to the image standard, the system measurements can closely resemble the manual measurements (higher accuracy). However, significant differences between system measurements and manual measurements can occur, as shown in Fig. 10 and Fig. 11, indicating that the system's accuracy will decrease, if the individual does not show the upper arm clearly and the color of the sleeve matches or resembles the background color.

Capturing images with a frontal viewing angle, proper distance, and adequate lighting tends to produce more accurate measurement results. This emphasizes the importance of quality and consistency in the image capture process to ensure that the automated measurement system can provide accurate and reliable results.

Evaluation using the Mean Absolute Error (MAE) metric resulted in an average error of 2.31 cm between system measurements and manual measurements. Of the 72 image samples tested, 21 individuals (29.2%) had measurement differences of 0-1 cm, 20 individuals (27.8%) had differences of 1-2 cm, and 31 individuals (43.1%) had differences of more than 2 cm. Thus, further development is expected to overcome the problems found in this study and improve the performance and accuracy of the MUAC detection and measurement system.

This study has several limitations. The dataset is modest and was collected under a standardized imaging protocol, which may limit generalizability. Pixel-to-centimeter calibration depends on a wall power-outlet reference that may not be available or standardized across sites. MUAC estimation fuses paired front- and side-view measurements via an ellipse approximation, making the final circumference susceptible to view-specific errors. The pipeline is also sensitive to clothing/background similarity and pose. Future work will expand data diversity and size, explore reference-free calibration, and strengthen the robustness of the segmentation and measurement pipeline.

ACKNOWLEDGMENT

This work was supported by "The DRTPM Kemendikbudristek of Republic Indonesia", under Research Grant 2023, in accordance with the Letter of Assignment Agreement, Number: 52/UN5.2.3.1/PPM/KP-DRTPM/B/2023.

REFERENCES

[1] C. Kyle and K. Jhon, Anthropometric Measurement. StatPearls Publishing, Treasure Island (FL), 2025. http://europepmc.org/books/NBK537315.

[2] B. G. Sisay, D. Haile, H. Y. Hassen, and S. H. Gebreyesus, "Performance of mid-upper arm circumference as a screening tool for identifying adolescents with overweight and obesity," PLoS ONE, vol. 15, no. 6, pp. 1–13, 2020, https://doi.org/10.1371/journal.pone.0235063.

[3] L. Thorup et al., "Mid-upper arm circumference as an indicator of underweight in adults: a cross-sectional study from Nepal," BioMed Central Public Health, vol. 20, no. 1, pp. 1–7, 2020, https://doi.org/10.1186/s12889-020-09294-0.

[4] E. Van Tonder et al., "Mid-upper arm circumference (MUAC) as a feasible tool in detecting adult malnutrition," South African Journal of Clinical Nutrition, vol. 32, no. 4, pp. 93–98, 2019, https://doi.org/10.1080/16070658.2018.1484622.

[5] A. Fakier, G. Petro, and S. Fawcus, "Mid-upper Arm Circumference : A Surrogate for Body," South African Medical Journal, vol. 107, no. 7, pp. 606–610, 2017, https://doi.org/10.7196/SAMJ.2017.vi7.12255.

[6] Febri Adriati and Shanty Chloranyta, "Status Gizi Ibu Hamil Berdasarkan Pengukuran Lingkar Lengan Atas (LILA)," Jurnal Kesehatan Panca Bhakti Lampung, vol. 10, no. 2, p. 127, 2022, https://doi.org/10.47218/jkpbl.v10i2.194.

[7] K. G. Mishra, V. Bhatia, and R. Nayak, "Association between mid-upper arm circumference and body mass index in pregnant women to assess their nutritional status," Journal of Family Medicine and Primary Care, , 2020, https://doi.org/10.4103/jfmpc.jfmpc_57_20

[8] D. T. Hendrixson, P. N. Lasowski, A. S. Koroma, and M. J. Manary, "Newborn Mid–Upper Arm Circumference Identifies Low–Birth Weight and Vulnerable Infants: A Secondary Analysis," vol. 6, no. 10, p. nzac138, 2022, https://doi.org/10.1093/cdn/nzac138.

[9] G. R. Babu et al., "Mid-upper arm circumference in pregnant women and birth weight in newborns as substitute for skinfold thickness: findings from the MAASTHI cohort study, India," BioMed Central Pregnancy and Childbirth, vol. 21, no. 1, pp. 1–11, 2021, https://doi.org/10.1186/s12884-021-03915-1.

[10] R. Rana et al., "Mid-Upper Arm Circumference Tapes and Measurement Discrepancies: Time to Standardize Product Specifications and Reporting," Global Health Science and Practice, vol. 9, no. 4, pp. 1011–1014, 2021, https://doi.org/10.9745/GHSP-D-21-00273.

[11] N. N. Kaashki et al, "Deep Learning-Based Automated Extraction of Anthropometric Measurements from a Single 3-D Scan," IEEE Transactions on Instrumentation and Measurement, vol. 70, pp. 1–14, 2021, https://doi.org/10.1109/TIM.2021.3106126.

[12] S. L. Cichosz et al., "Precise Prediction of Total Body Lean and Fat Mass From Anthropometric and Demographic Data: Development and Validation of Neural Network Models," Journal of Diabetes Science and Technology, vol. 15, no. 6, pp. 1337–1343, 2020, https://doi.org/10.1177/1932296820971348.

[13] Š. Dana, R. Adam, and M. Martin, "Automatic Estimation of Anthropometric Human Body Measurements," VISAPP, 2022, vol. 4, no. 1, pp. 537–544, https://doi.org/10.5220/0010878100003124.

[14] M. Edoardo et al., "Digital Anthropometry: A Systematic Review on Precision, Reliability and Accuracy of Most Popular Existing Technologies," Nutrients, vol. 15, no. 2, pp. 1–39, 2023, https://doi.org/10.3390/nu15020302.

[15] D. S. Bukit, M. S. Lydia, P. I. Nainggolan, H. I. Mahmud, R. M. Sembiring, R. M. Hasibuan, D. Athirah, F. M. Mufida, and R. W. B. O. K. Rahmat, "Leveraging Machine Learning Techniques for Stunting Detection and Height Growth Prediction in Children Aged 0–5 Years," ELTICOM 2024. pp. 130–134, 2024, https://doi.org/10.1109/ELTICOM64085.2024.10864967.

[16] Yingying Liu, Arcot Sowmya, and Heba Khamis, "Single camera multi-view anthropometric measurement of human height and mid-upper arm circumference using linear regression," PLoS ONE, vol. 13, no. 4, pp. 1–22, 2018, https://doi.org/10.1371/journal.pone.0195600.

[17] H. Qinhua et al., "An effective approach for CT lung segmentation using mask region-based convolutional neural networks," Artificial Intelligence in Medicine, vol. 103, p. 101792, 2020, https://doi.org/10.1016/j.artmed.2020.101792.

[18] M. Fan et al., "Mass Detection and Segmentation in Digital Breast Tomosynthesis Using 3D-Mask Region-Based Convolutional Neural Network: A Comparative Analysis," Frontiers in Molecular Biosciences, vol. 7, pp. 1–15, 2020, https://doi:10.3389/fmolb.2020.599333.

[19] Linqin Cai, Tao Long, Yuhuan Dai, and Yuting Huang, "Mask R-CNN-Based Detection and Segmentation for Pulmonary Nodule 3D Visualization Diagnosis," IEEE Access, vol. 8, pp. 44400–44409, 2020, https://doi:10.1109/ACCESS.2020.2976432.

[20] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick, “Mask R-CNN” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 42, no .2, pp. 386–397, 2020, https://doi.org/10.1109/TPAMI.2018.2844175.

[21] K. Alexander et al., “Panoptic Segmentation,” in 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 9396–9405. https://doi.org/10.1109/CVPR.2019.00963.

[22] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137-1149, 1 June 2017. https://doi.org/10.1109/TPAMI.2016.2577031

[23] Lin, TY. et al. (2014). Microsoft COCO: Common Objects in Context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds) Computer Vision – ECCV 2014. vol 8693. Springer, Cham. https://doi.org/10.1007/978-3-319-10602-1_48

[24] T. -Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan and S. Belongie, "Feature Pyramid Networks for Object Detection," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017, pp. 936-944, https://doi.org/10.1109/CVPR.2017.106

[25] F. Hou et al., “Improved Mask R-CNN with distance guided intersection over union for GPR signature detection and segmentation,” Autom Constr, vol. 121, p. 103414, Jan. 2021, https://doi.org/10.1016/J.AUTCON.2020.103414.

[26] Botchkarev, A. (2019). A new typology design of performance metrics to measure errors in machine learning regression algorithms. Interdisciplinary Journal of Information, Knowledge, and Management, 14, 45–76. https://doi.org/10.28945/4184.

[27] B. Dwyer, J. Nelson, and T. Hansen, “Roboflow,” 2024, 1.0. Accessed: Apr. 20, 2025. [Online]. Available: https://roboflow.com

[28] Wu, Yuxin, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. "Detectron2." (2019): https://github.com/facebookresearch/detectron2.

[29] M. A. et al., “Accuracy of Fully Automated 3D Imaging System for Child Anthropometry in a Low-Resource Setting: Effectiveness Evaluation in Malakal, South Sudan,” JMIR Biomed Eng, vol. 7, no. 2, p. e40066, Oct. 2022, https://doi.org/10.2196/40066.

Ensemble Learning for Multi-Class Android Malware Detection: A Robust Framework for Family Level Classification

Mana Saleh Al Reshan
Department of Information System-College of Computer Science and Information Systems, Najran University, Najran 61441, Saudi Arabia
Emerging Trends and Technologies Lab (ETRL)-College of Computer Science and Information Systems, Najran University, Najran 61441, Saudi Arabia

***Abstract*—The widespread popularity of Android devices has made them a prime target for sophisticated and evolving malware threats. Traditional malware detection techniques rely on binary classification (malicious vs. benign), which fails to capture the nuanced behavioral differences between malware families, critical for threat intelligence and incident response. To address this limitation, we propose a robust multi-class classification approach for Android malware family detection, leveraging ensemble learning and advanced feature selection methods. Our system uses a hybrid feature extraction strategy that combines Chi-Squared and Mutual Information techniques to eliminate low-utility features and retain the most discriminative attributes. These include flow-based metrics, inter-arrival time (IAT), and session duration, key indicators of malicious behavior. We evaluated five baseline classifiers (Random Forest, Gradient Boosting, XGBoost, Extra Trees, and Decision Trees) across three ensemble strategies (bagging, voting, and stacking). Among these, the Stacking ensemble achieved the highest overall performance, with 83% across all evaluation metrics, accuracy, precision, recall, and F1-score, and a True Negative Rate (TNR) of 93.34%. The framework also improves the detection of minority malware families in imbalanced datasets. These findings highlight the advantages of ensemble learning for building scalable and reliable Android systems suitable for real-world deployment.**

***Keywords*—Malware detection; cyber threat; ML models; feature selection; ensemble methods**

I. Introduction

The rapid proliferation of intelligent mobile devices has established Android as the dominant smartphone computing platform. In 2020, global smartphone production reached approximately 1.24 billion units, with Android accounting for around 78.45% of the market share [1]. This dominance can be attributed to its open-source architecture, affordability, and extensive connectivity options, including Wi-Fi, Bluetooth, and GPS. Additionally, Android's support for third-party application installation significantly enhances device functionality.

However, this openness introduces critical security vulnerabilities. Unlike Apple's tightly controlled ecosystem, Android's decentralized app distribution model facilitates the spread of malicious applications beyond the official Google Play Store. Despite Google's integrated malware scanning systems, many harmful apps still manage to evade detection [2]. The fragmented nature of the Android ecosystem further compounds security challenges, increasing exposure to threats such as data breaches and system-level attacks. In the third quarter of 2021, the Kaspersky Security Network reported over 9.5 million blocked incidents involving mobile malware, adware, and riskware. Common infection vectors include malicious MMS messages, phishing emails, and compromised applications. Users are particularly at risk when installing unverified apps that embed malicious code or exploit system vulnerabilities [3]. With Android powering approximately 70% of global mobile devices, it remains a prime target for various cyberattacks, including malware, ransomware, and spyware. Frequently used apps like WhatsApp, Facebook, mobile banking tools, and games are often exploited, placing user data such as photos, contacts, messages, and financial information at significant risk. This underscores the urgent need for intelligent and reliable malware detection systems to safeguard user privacy and device security.

Machine learning (ML), particularly ensemble learning techniques, has shown great promise in malware classification [4]. These methods enhance detection accuracy and offer robustness against evasion techniques. Ensemble techniques such as bagging, voting, and stacking combine the predictive power of multiple classifiers, mitigating the weaknesses of individual models and improving overall performance [5].

Unlike traditional single-classifier methods, such as Decision Trees or Gradient Boosting, which are prone to overfitting and poor generalization on large and varied malware datasets, our ensemble learning methodology provides tangible enhancements. For example, in a stacking ensemble research on the AndMal 2020 dataset, the authors obtained 83% accuracy and a TNR of 93.4% for multi-class classification, which beat traditional models hands down. Similarly, recent deep-learning-based ensembles combining CNN variants (ResNet, SENet, SEResNet) in "MFEMDroid" have also yielded superior detection performance owing to their complementary strengths. These results further substantiate that ensemble methods, specifically stacking, bagging, and voting, not only enhance detection precision and reduce false positives but also are more robust against evasive

attacks. Further, in contrast to most existing research centering on binary classification only, our framework contributes more by allowing multi-family detection (Adware, Spyware, SMS_Malware, Benign), enabling richer threat identification and improved real-world applicability.

This research proposes and evaluates an Android malware detection framework that employs various ensemble learning methods to classify malware samples into families. The primary objective is to enhance detection reliability, reduce false positives, and bolster mobile cybersecurity.

Given the widespread use of Android smartphones, this study addresses a critical challenge in mobile security. By applying ensemble learning to classify malware into families, the proposed framework facilitates early threat identification and contributes to the development of intelligent, automated defense mechanisms. The findings are valuable to researchers, cybersecurity professionals, and mobile developers aiming to enhance the security of mobile environments.

Key Contributions:

- We obtained an Android malware dataset from the Kaggle repository. We utilized the Android Malware Dataset from Kaggle, consisting of 355,630 records with 85 features. Categorized into four families: Spyware, Adware, SMS_Malware, and Bengin.
- The dataset was preprocessed by handling null values, removing duplicates and irrelevant data, and applying label encoding to categorical features.
- Feature selection was performed using Chi-square and Mutual Information (MI) techniques to retain the most relevant features.
- We trained and evaluated five baseline machine learning classifiers: Random Forest (RF), Gradient Boosting (GB), Extreme Gradient Boosting (XGB), Extra Trees (ET), and Decision Tree (DT). In addition, we implemented and assessed three ensemble methods: stacking, bagging, and voting.
- This study presents a reproducible framework that can serve as a foundation for future research in mobile cybersecurity and intelligent malware detection.

The remainder of the study is structured as follows: Section II reviews related work on Android malware and Machine learning approach. Section III describes the dataset, preprocessing steps, and feature selection used in this study. Section IV presents the experimental results and provides comparative studies. Section V concludes the study and outlines the future direction.

II. Related Work

An algorithm for malware detection was proposed in this work [6]. Two Android malware datasets, including CICAndMal2017 and Drebin, were selected for conducting research. To detect malware attacks, the researchers implemented support vector machine (SVM), K-nearest neighbors (KNN), Linear Discriminant Analysis (LDA), Long Short Term Memory (LSTM), and Conventional Neural Network-Long Short Term Memory (CNN-LSTM). The highest accuracy rate, which is 100% was obtained by the SVM model using the CICAndMal2017 dataset.

This research conducted a systematic literature review to examine deep learning for malware attack detection. Their research spans 2014 to 2021, and they identified a total of 123 studies. They found 40.1% (53) of studies were deep learning-based. To fulfill their goal, some research questions were established [7].

Malicious pattern analysis can help reverse the misuse of online applications, such as banking and various social media apps. In study [8], an IOS malware categorization model was proposed, which is capable of detecting 30 patterns of phylogenetic. Out of 150mobile apps, only 7 apps were identified by the classifier.

Android Malware grayscale image representation visualization methods were proposed [9]. The authors utilize both global and local features for better performance. It was observed that local features such as manifest, dex-arc, produce poor accuracies between 65.16% and 93.56%. However, orb features take low computational time for feature selection, training extraction model training, and malware sample detection.

With the growing popularity of Android smartphones, malware threats are receiving increased attention. This study [10] introduces MLDroid, a web-based framework for detecting Android malware through dynamic analysis. MLDroid identifies relevant features using feature selection techniques and trains models using various machine learning approaches. The system was evaluated on over 500,000 real-world apps. Results show that integrating four techniques, such as deep learning, Farthest First Clustering, Y-MLP, and a nonlinear ensemble decision tree forest, along with rough set-based feature selection, achieved a high malware detection rate of 98.8%.

The growing use of Android devices has led to the emergence of sophisticated malware that employs complex obfuscation techniques, making static analysis alone inadequate for effective detection. Although dynamic analysis can uncover evasive behaviors, it is resource-intensive and limited in execution path coverage. To overcome these limitations, a hybrid detection approach combining static and dynamic analysis was proposed. A publicly available dataset containing 352 static and 323 dynamic features was developed to support both binary and multiclass classification. Irrelevant features were removed using the Information Gain algorithm. Experimental results demonstrated that the hybrid method significantly outperformed individual analyses, with the Random Forest classifier achieving 98.53% accuracy for binary classification and 90.1% for multiclass classification [11].

Android malware is a persistent and growing threat, affecting industries like healthcare, banking, transportation, government, and e-commerce. It became challenging to classify malicious apps. This study [12] proposed two ML-enabled dynamic analysis methods: one to categorize malware by category and the other by family. With the CCCS-CIC-

AndMal2020 dataset that contains 14 malware categories and 180 families, the proposed models were more than 96% accurate in detecting categories and more than 99% accurate in detecting families.

As Android devices gain popularity, malware detection and family categorization have become the most important research topics. Although static and dynamic analysis are most commonly used, they are based on complicated processes. In this work [13], the researchers present a hybrid approach optimized for multi-feature data to detect and categorize Android malware. Static analysis makes use of permissions and intents, three feature selection techniques being tested, with the best results from Chi-Square, yielding a 95.04% detection rate using a Random Forest classifier. Main static features were also tested for more information. In dynamic analysis, as opposed to previous methods involving one-way HTTP or transport-layer traffic, the researchers maintained complete session-level protocol layers. The Res7LSTM model was subsequently used to further classify uncertain samples. Experimental results indicate that the proposed method enhances malware family classification accuracy from 71.48% (previous work) to 99% by taking advantage of session-based, multi-layer traffic analysis with high accuracy using fewer static features.

Malware family classification categorizes similar malware instances into families to determine their behavioral patterns and facilitate recovery. A permission-based classification technique with both native and native/custom Android permissions is proposed [14]. Experiments were performed on the DREBIN dataset with various classifiers and tested using accuracy, macro F1-score, balanced accuracy (BAC), and Matthews correlation coefficient (MCC), the latter two measures being appropriate for imbalanced datasets. The results indicated that: i) tailor-made permissions enhanced classification performance, ii) accuracy and BAC varied up to 3.67% despite having the same model, and iii) LightGBM and AdaBoost performed better than other classifiers.

The detection and categorization of Android malware families must be automated to reduce the workload of security analysts. While various machine learning and deep learning techniques exist, the rapid growth of mobile applications necessitates faster and more accurate solutions. This study proposes a scalable method that significantly improves both detection accuracy and analysis speed. Evaluated on large-scale datasets, the proposed approach achieves an F-measure of 99.71%, a low false positive rate of 0.37%, and processes 300,000 samples in just 3.3 hours. For malware family classification across 28 families, it achieves an F-measure of 97.5%, a precision of 96.55%, and a recall of 98.64%. Comparative results demonstrate that our approach outperforms previous methods in both efficiency and effectiveness [15].

III. Research Methodology

This section presents a detailed description of the entire pipeline employed for Android Malware families' classification, comprising dataset preprocessing, feature selection, data splitting, training and testing of baseline classifiers, and the ensemble methods, as shown in Fig. 1.

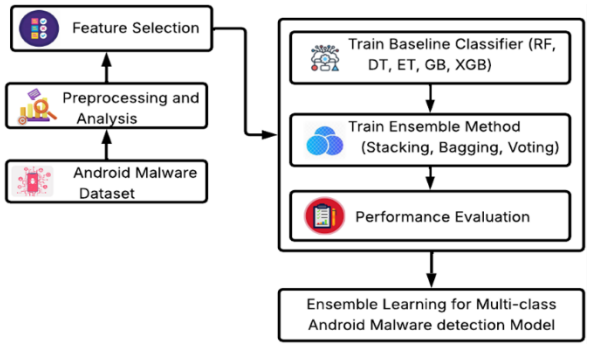


Fig. 1. Proposed research methodology.

A. Dataset

We downloaded the Android Malware dataset from Kaggle [16]. The dataset was comprised of (355630 x 85) records. The dataset contains four families, such as Adware, Spyware, SMS_malware, and Bengin, as depicted in Fig. 2.

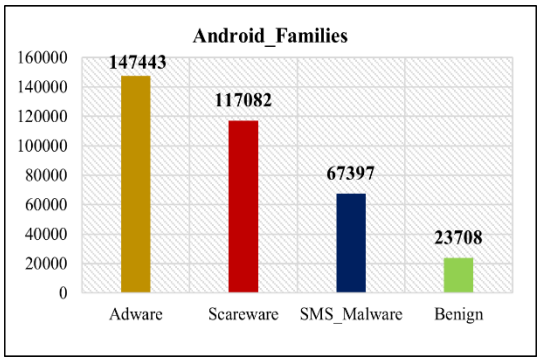


Fig. 2. Android families.

B. Preprocessing and Analysis

A non-informative timestamp column was removed, as it did not contribute any meaningful value to the training process. To prepare the dataset for machine learning algorithms, categorical features were encoded using the Label Encoding technique. Each categorical column was converted into a numerical format by assigning a unique integer to every distinct category. Missing values were either filled or removed, as appropriate, to maintain consistency and data integrity. After encoding, the dataset was separated into feature variables (X) and target labels (y). A final check for missing values in the feature set was conducted to ensure data completeness before model training and feature selection. These preprocessing steps ensured that the data was clean, fully numeric, and suitable for both statistical analysis and machine learning applications [17].

C. Feature Selection

To enhance model performance and reduce computational complexity, feature selection was performed using two statistical techniques: the Chi-Squared Test and Mutual Information.

1) Chi-Square test: The Chi-Squared test is a statistical hypothesis test used to assess the independence between categorical features (attributes) and the target class. In the context of feature selection, it assesses whether the actual

frequency distribution of a feature significantly differs from the expected distribution assuming that the feature is statistically independent of the target variable [18]. Mathematically, the Chi-Squared statistic for a feature f is defined, as shown in Eq. (1):

$$Z^2 = \sum_{i=1}^{t} \sum_{k=1}^{d} \frac{(M_{ik} - P_{ik})^2}{P_{ik}} \quad (1)$$

where,

M_{ik}: Signifies the observed frequency of category i of features f with class labels k

P_{ik} characterizes the expected count of frequency assuming independence.

$\boldsymbol{k}$: Depicts the number of feature categories

$\boldsymbol{d}$: Shows the number of class labels

Fig. 3 demonstrates the selected features by their Chi-Squared scores, highlighting their relevance for classification. Top features such as Flow Bytes/s, Flow IAT Std, and various Inter-Arrival Time (IAT) metrics are essential for distinguishing normal traffic from anomalies or attacks. The prominence of IAT-related features (e.g., Fwd IAT Total, Bwd IAT Mean) underscores the importance of packet timing in detecting abnormal patterns linked to malicious activity. Metrics like Flow Bytes/s and Packet Length Variance help identify unusual data rates and packet sizes, often seen in network intrusions. Active and Idle time features further reveal session-level behavior, aiding in the detection of stealthy threats. The high Chi-Squared values (up to 10th) reflect the strong discriminatory power of these features. These statistically selected features provide a solid foundation for enhancing intrusion detection and traffic classification through machine learning.

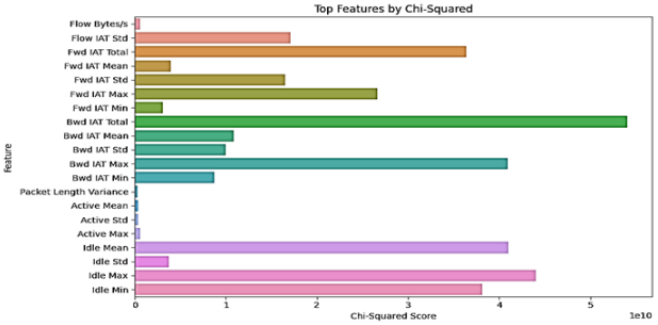


Fig. 3. Selected features using the Chi-Square method.

To enhance feature selection beyond the limitations of Chi-Squared analysis, Mutual Information (MI) was employed to capture both linear and non-linear dependencies between features and the target variable. This approach helps optimize the feature set by removing less informative attributes, particularly in scenarios involving complex network traffic patterns.

2) Mutual information feature selection: Mutual Information (MI) was used to quantify the amount of shared information between each feature and the target variable. Unlike the Chi-Squared test, MI can capture both linear and non-linear relationships. MI scores were computed for all features, and the top 20 with the highest scores were selected [19].

Fig. 4 presents a hierarchical listing of network traffic features sorted by their Mutual Information scores, indicating their predictive relevance to the target variable (e.g., intrusion detection). The top-ranked feature is Flow ID, followed by key metrics such as Fwd Avg Bytes/Bulk, Packet Length Mean, and Avg Bwd Segment Size, each with related sub-features indented beneath. These attributes represent various aspects of network behavior, including flow identifiers, packet size distributions, data transfer characteristics, and traffic dynamics. A supporting (though currently blank) tabular structure appears to outline threshold-based MI scores, likely intended to guide the selection of highly informative features. This step plays a critical role in building more accurate and robust machine learning models for network traffic classification and security analysis.

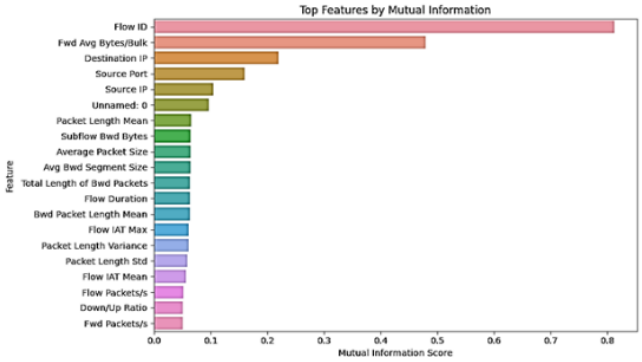


Fig. 4. Selected features using the MI method.

D. Model Development

We trained and evaluated four baseline classifiers, such as RF, ET, GB, XGB, and DT. To enhance predictive accuracy and robustness, we also implemented ensemble methods including stacking, bagging, and voting. Algorithm 1 demonstrates the training and testing mechanism of baseline and ensemble methods.

Algorithm-1: Training and Evolution of Baseline Classifiers

Input: Selected features set
Output: evaluation metrics for each baseline classifier
Begin:

1. Split the dataset into features (X) and Target label (Y)
2. Apply train_test_split on X and Y with stratification
3. Define baseline classifiers
   - RF
   - DT
   - ET
   - GB
   - XGB
4. For each baseline classifier:
   - Train model
   - Compute evaluation metrics:
     - Accuracy, precision, recall, F2 score, TNR
     - Confusion matrix
5. Train ensemble methods on trained baseline classifiers
6. Repeat Step 4 for each ensemble method

end

IV. Results and Discussion

The experiments in this study were performed on Windows 10 using Kaggle Notebook. The dataset was divided into training (70%) and test (30%) sets, with stratification to maintain the original label distribution, as depicted in Table I below.

TABLE I Division of Samples

Training Samples	Testing samples	Total Samples
248938	106688	355626

TABLE II Performance Evaluation of ML Models

ML models	Accuracy	Precision	Recall	F1 score	TNR
RF	81%	81%	81%	80%	92.37%
GB	70%	70%	70%	66%	87.95%
ET	79%	79%	79%	79%	91.72%
DT	79%	79%	79%	79%	91.92%
XGB	77%	77%	77%	77%	91.05%
Stacking	83%	82%	80%	8%	93.34%
Bagging	80%	89%	80%	80%	92.13%
Voting	82%	82%	82%	82%	92.95%

Table II presents the comparative performance of various machine learning models evaluated on a multi-class classification problem. Among the evaluated models, the Stacking classifier demonstrates the best overall performance, achieving 83% across accuracy, precision, recall, and F1 score, along with the highest true negative rate (TNR) of 93.34%. This reflects its strong generalization and balanced class discrimination, likely due to its ability to aggregate diverse base learners and optimize their combined predictions. Voting (82%) and RF (81%) also perform well across all metrics, benefiting from ensemble techniques that reduce variance and enhance stability. Bagging, while showing slightly lower accuracy (80%), achieves the highest precision (89%), indicating confident positive predictions, albeit with a trade-off in recall. ET (79%), DT (79%), and XGBoost (77%) yield moderate, balanced results but are outperformed by more advanced ensemble methods. GB exhibits the weakest performance, with 70% accuracy and the lowest F1 score (66%), possibly due to overfitting or suboptimal parameter tuning. Overall, ensemble approaches, especially stacking and voting, offer superior predictive performance and robustness, while single models and simpler boosting methods lag in capturing complex class relationships.

TABLE III Cross-Validation

ML models	Accuracy (CV =5)	Mean Std
RF	87%	± 0.0008
GB	88%	± 0.0011
ET	86%	± 0.0007
DT	83%	± 0.0007
XGB	84%	± 0.0009
Stacking	89%	± 0.0003
Bagging	84%	± 0.0009
Voting	85%	± 0.0012

To ensure the robustness of the ML model, 5-fold stratified cross-validation was performed. Table III describes the performance of ML classifiers and ensemble methods. The analysis illustrated that ensemble methods outperformed individual learners consistently. Among the baseline classifiers, GB (88%), RF (87%), and ET (86%) demonstrated robust predictive accuracy, while the lone DT ranked lower at 83% because of its greater overfitting tendency. XGBoost (84%) performed competitively, but could not surpass GB within this dataset. Ensemble methods improved results even more, with the highest accuracy achieved by the Stacking Classifier (89%) and lowest variance (±0.0003), performing superior generalization through diverse base learners via a meta-model. Bagging (84%) and Voting (85%) produced incremental improvements over single classifiers, but were inferior to stacking. The uniformly low standard deviations between models reflect consistently stable and dependable performance, and stacking proves to be the most stable and dependable means for deployment.

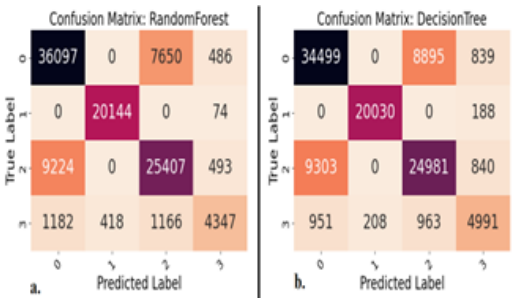


Fig. 5. Confusion matrix for ML models.

Fig. 5 represents the performance of the RF and DT models over unseen test data using confusion matrices. In Fig. 5(a), the RF model correctly classified 85995 out of the total samples (106688) with 20693 misclassifications, demonstrating high predictive accuracy. Similarly, Fig. 5(b) shows the DT model's confusion matrix, with 84501 correct predictions and 22187 misclassifications. The superior performance of the RF model compared to the DT model is expected, given RF's ensemble nature. By combining the predictions of multiple decision trees through bootstrap aggregation (bagging) and random feature selection, RF reduces variance and improves generalization to unseen data. In contrast, a single DT is more prone to overfitting, often capturing noise in the training set. The confusion matrix results further demonstrate that the RF model more effectively handles class boundaries, resulting in fewer misclassifications than the DT model.

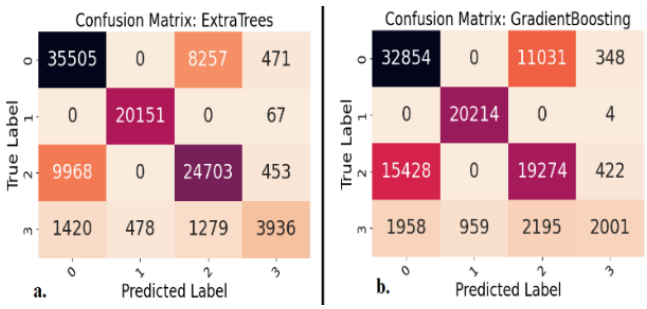


Fig. 6. Performance evaluation of classifiers.

Fig. 6(a) presents the confusion matrix for the ET classifier. The model demonstrated high predictive accuracy for class 1 and class 2, with 20,151 and 24,703 correct predictions, respectively. However, it showed significant misclassification between class 0 and class 2: 8,257 class 0 instances were incorrectly predicted as class 2, and 9,968 class 2 instances were misclassified as class 0, indicating notable overlap between these two classes. Fig. 6(b) shows the confusion matrix for the GB model. It correctly identified 20,214 class 1 and 19,274 class 2 samples, but exhibited greater confusion between class 0 and class 2 than Extra Trees, misclassifying 15,428 class 2 instances as class 0 and 11,031 class 0 instances as class 2. Both models performed poorly on class 3, with substantial dispersion across incorrect classes, indicating limited discriminative power for this class. Both models consistently underperform on class 3, with predictions for this class dispersed across the other categories. This suggests that class 3 may be underrepresented in the training data or lacks sufficiently distinctive features, making it difficult for the models to learn clear decision boundaries. The consistently poor classification performance for class 3 underscores a fundamental limitation in the dataset, either in terms of class imbalance or inadequate feature representation for this class.

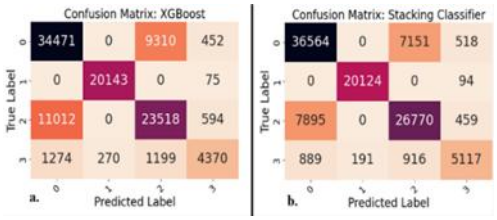


Fig. 7. Performance evaluation of ML models over unseen data.

Fig. 7(a) and Fig. 7(b) compare the performance of XGBoost and the Stacking classifier. XGBoost performs well on classes 1 and 2, with 20,143 and 23,518 correct predictions, respectively, as depicted in Fig. 7(a). However, in Fig. 7(b), it shows significant confusion between class 0 and class 2, misclassifying 9,310 class 0 instances as class 2 and 11,012 class 2 instances as class 0. This indicates overlapping feature space or weak decision boundaries between these classes. In contrast, the Stacking classifier demonstrates stronger overall performance. It substantially increases correct predictions for class 0 (36,564) and class 2 (26,770), while reducing misclassifications between them. Additionally, it improves classification for class 3, correctly identifying 5,117 instances, suggesting better generalization and improved handling of minority classes through ensemble learning. Overall, the Stacking model achieves more balanced classification and effectively addresses the limitations observed in XGBoost.

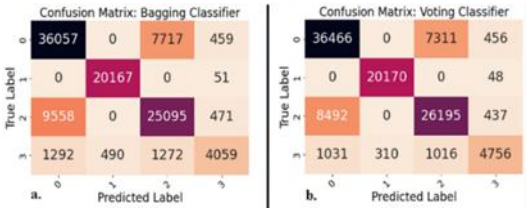


Fig. 8. Confusion matrix for ensemble methods.

The confusion matrices compare the bagging and voting classifiers. Fig. 8(a) shows the confusion matrix for the Bagging Classifier, where the classifier is showing high accuracy for class 0 (36,057 correct predictions) with fewer overall misclassifications. The Voting Classifier slightly improves correct predictions for class 0 (36,466) but introduces greater confusion, notably misclassifying 7,311 class 0 instances as class 1, as represented in Fig. 8(b). Diagonal entries indicate correct classifications, while off-diagonal entries represent errors. Bagging uses bootstrap aggregation to reduce variance and produce stable predictions. In contrast, voting combines outputs from diverse models, enhancing generalization but sometimes increasing misclassification between closely related classes. These results illustrate the trade-off between model diversity and predictive stability in ensemble methods.

The baseline classifier comparison provides key insights into performance across standard evaluation metrics, as represented in Fig. 9. Among the five models, RF, GB, ET, DT, and XGB. The XGBoost demonstrates the strongest performance, achieving the highest accuracy (92.37%) along with solid precision (81%), recall (81%), and F1 score (80%). This is largely due to its advanced regularization, gradient-based optimization, and ability to capture complex patterns while mitigating overfitting. All ensemble methods (RF, GB, ET, and XGB) outperform the single Decision Tree, which shows the lowest accuracy (87.95%) and reduced precision, recall, and F1 scores (all around 70%). This highlights the benefit of ensemble learning in enhancing generalization by combining multiple learners. ET (91.92%) and Random Forest (91.05%) also perform well, leveraging random feature selection and ensemble averaging to improve robustness. GB (91.72%), while slightly behind XGBoost, still delivers strong performance but lacks some of XGBoost's advanced optimization features, which likely explains the gap. In summary, XGBoost emerges as the most effective classifier, though other ensemble methods remain competitive. While the Decision Tree model is simple and interpretable, it lacks the predictive power needed for applications requiring high accuracy.

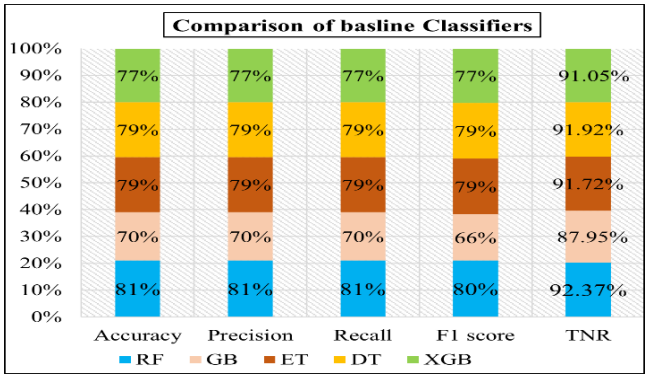


Fig. 9. Comparison of ML classifiers.

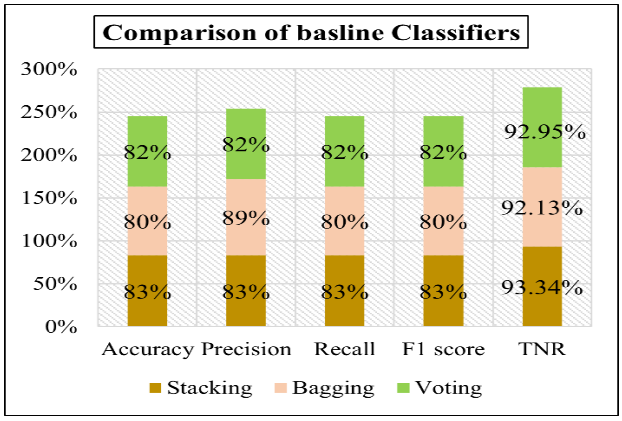


Fig. 10. Comparison of ensemble methods.

Fig. 10 depicts the comparison of three ensemble methods to determine the most effective approach. Key metrics, including accuracy, precision, recall, F1 score, and true negative rate (TNR), are used to evaluate performance, with values such as 92.95%, 92.13%, and 93.34% indicating strong results in individual configurations. Ensemble techniques, namely stacking, bagging, and voting, are also assessed for their ability to enhance model robustness by combining multiple classifiers. The inclusion of varying percentages (e.g., 30%, 25%, 20%) suggests an investigation into data split ratios or classifier weighting, aimed at further optimizing performance. Consistent precision values, such as 82%, 80%, and 83% across different ensemble methods, highlight the stability and reliability of these approaches. Overall, the analysis underscores the importance of selecting appropriate evaluation metrics and ensemble strategies to maximize classifier accuracy and predictive performance.

A. Discussion

The increased popularity of Android phones made them a favorite target among malicious applications, thus rapidly increasing malware on Android. Traditional detection systems typically perform binary classification (malicious vs. benign), overlooking the nuanced distinctions between malware families and differences that are crucial for threat intelligence, incident response, and forensic analysis [20] [21]. Addressing this gap, our study introduces a multi-class classification framework to differentiate between Android malware families using ML and ensemble-based models. As indicated in Table II, ensemble methods, specifically stacking, consistently outperformed baseline standalone models. The Stacking classifier obtained 83% in all the evaluation metrics (accuracy, precision, recall, F1-score) and a maximum TNR of 93.34%, validating its better generalization ability. Confusion matrix analyses also reveal the virtues and vices of separate classifiers. Models such as Random Forest and Extra Trees performed well with high accuracy for dominant families (e.g., Class 1 and Class 2), but struggled with minority classes (e.g., Class 3), revealing an ongoing class imbalance problem. Interestingly, Stacking demonstrated better performance with underrepresented classes, correctly classifying 5,117 examples of Class 3 and illustrating that it was capable of detecting intricate, non-linear relationships. Feature selection had an essential impact on model performance. Employing a hybrid technique incorporating Chi-Squared and Mutual Information algorithms, we found discriminative features like Flow Bytes/s, inter-arrival time (IAT) metrics, and session-based duration corresponding to behavioral patterns of various malware families. The dual-method strategy allowed dimensionality reduction to be done in an efficient manner without losing interpretability and enhancing model accuracy. Decision Trees were the worst-performing classifiers among all because of overfitting and the poor learning capacity of complex data patterns. Conversely, Voting classifiers and XGBoost yielded close-to-optimal performance, further supporting the efficacy of aggregation and boosting on high-variance, imbalanced datasets, such as Android malware. Ensemble learning supplemented with careful feature selection is an effective approach to steady multi-class Android malware detection, especially in real-world environments where data imbalance and intricate decision boundaries are prevalent.

V. Conclusion

This work presents an efficient and scalable machine learning architecture for multi-class Android malware family classification, addressing the limitations of traditional binary detection approaches. By leveraging ensemble learning, specifically the stacking method and a hybrid feature selection strategy combining Chi-Squared and Mutual Information techniques, the system achieves high accuracy, strong generalizability, and robustness to data imbalance. Empirical results demonstrate that the Stacking classifier outperforms both individual models and other ensemble techniques, achieving 83% across all evaluation metrics and the highest TNR of 93.34%. Notably, it significantly improves the detection of minority malware families, making it well-suited for real-world threat environments characterized by imbalanced class distributions and complex behavioral patterns. These findings validate the effectiveness of ensemble-based approaches and principled feature engineering in malware detection and provide a solid foundation for advancing automated and intelligent mobile security systems. Future work will focus on expanding the dataset, incorporating dynamic analysis features, and integrating deep learning models to further enhance detection accuracy and threat response capabilities.

VI. Future Work

While the present study demonstrates the efficacy of ensemble learning for Android malware detection, various promising research directions remain unexplored. To begin with, expanding the dataset to include newer and varied malware samples, such as zero-day attacks, would make the model more robust against changing threats. Second, the incorporation of dynamic analysis capabilities like system calls, network traffic, and runtime activity might supplement the static capabilities employed in this research to give a better understanding of malware activity. Third, combining ensemble techniques with deep learning frameworks (like CNNs, RNNs, or combination models) might further enhance detection rates, particularly in dealing with intricate temporal and behavioral patterns. Additionally, future efforts need to tackle adversarial robustness, as malware developers more and more use evasion and obfuscation mechanisms to defeat machine learning-based

detectors. Research on federated and distributed learning methods is also a promising direction, enabling collaborative malware detection on devices without compromising user privacy. Finally, using explainable AI (XAI) techniques would add more transparency and trust in detection outcomes, which is extremely essential for cybersecurity professionals when deploying tools in critical real-world settings.

REFERENCES

[1] M. Huang, "Press Center - Yearly 5G Smartphone Production Projected to Exceed 200 Million Units Thanks to Smartphone Brands' Proactive Push in 2H20, Says TrendForce | TrendForce - Market research, price trend of DRAM, NAND Flash, LEDs, TFT-LCD and green energy, PV," TrendForce. Accessed: Jul. 02, 2025. [Online]. Available: https://www.trendforce.com/presscenter/news/20200722-10398.html

[2] O. N. Elayan and A. M. Mustafa, "Android Malware Detection Using Deep Learning," Procedia Computer Science, vol. 184, pp. 847–852, Jan. 2021, doi: 10.1016/j.procs.2021.03.106.

[3] "IT threat evolution in Q3 2021. Mobile statistics." Accessed: Jul. 02, 2025. [Online]. Available: https://securelist.com/it-threat-evolution-in-q3-2021-mobile-statistics/105020/

[4] K. Liu, S. Xu, G. Xu, M. Zhang, D. Sun, and H. Liu, "A Review of Android Malware Detection Approaches Based on Machine Learning," IEEE Access, vol. 8, pp. 124579–124607, 2020, doi: 10.1109/ACCESS.2020.3006143.

[5] D. Gupta and R. Rani, "Improving malware detection using big data and ensemble learning," Computers & Electrical Engineering, vol. 86, p. 106729, Sep. 2020, doi: 10.1016/j.compeleceng.2020.106729.

[6] H. Alkahtani and T. H. H. Aldhyani, "Artificial Intelligence Algorithms for Malware Detection in Android-Operated Mobile Devices," Sensors, vol. 22, no. 6, Art. no. 6, Jan. 2022, doi: 10.3390/s22062268.

[7] Y. Liu, C. Tantithamthavorn, L. Li, and Y. Liu, "Deep Learning for Android Malware Defenses: A Systematic Literature Review," ACM Comput. Surv., vol. 55, no. 8, p. 153:1-153:36, Dec. 2022, doi: 10.1145/3544968.

[8] M. Mohd Saudi, M. A. Husainiamer, A. Ahmad, and M. Y. I. Idris, "iOS mobile malware analysis: a state-of-the-art," J Comput Virol Hack Tech, vol. 20, no. 4, pp. 533–562, Nov. 2024, doi: 10.1007/s11416-023-00477-y.

[9] H. M. Ünver and K. Bakour, "Android malware detection based on image-based features and machine learning techniques," SN Appl. Sci., vol. 2, no. 7, p. 1299, Jun. 2020, doi: 10.1007/s42452-020-3132-2.

[10] A. Mahindru and A. L. Sangal, "MLDroid—framework for Android malware detection using machine learning techniques," Neural Comput & Applic, vol. 33, no. 10, pp. 5183–5240, May 2021, doi: 10.1007/s00521-020-05309-4.

[11] M. Dhalaria and E. Gandotra, "A hybrid approach for Android malware detection and family classification," IJIMAI, vol. 6, no. 6, pp. 174–188, 2021.

[12] A. H. E. Fiky, A. E. Shenawy, and M. A. Madkour, "Android Malware Category and Family Detection and Identification using Machine Learning," Jul. 05, 2021, arXiv: arXiv:2107.01927. doi: 10.48550/arXiv.2107.01927.

[13] C. Ding, N. Luktarhan, B. Lu, and W. Zhang, "A Hybrid Analysis-Based Approach to Android Malware Family Classification," Entropy, vol. 23, no. 8, Art. no. 8, Aug. 2021, doi: 10.3390/e23081009.

[14] M. Kim, D. Kim, C. Hwang, S. Cho, S. Han, and M. Park, "Machine-Learning-Based Android Malware Family Classification Using Built-In and Custom Permissions," Applied Sciences, vol. 11, no. 21, Art. no. 21, Jan. 2021, doi: 10.3390/app112110244.

[15] B. Sun, T. Takahashi, T. Ban, and D. Inoue, "Detecting Android Malware and Classifying Its Families in Large-scale Datasets," ACM Trans. Manage. Inf. Syst., vol. 13, no. 2, p. 12:1-12:21, Oct. 2021, doi: 10.1145/3464323.

[16] "Android Malware Detection." Accessed: Jul. 02, 2025. [Online]. Available: https://www.kaggle.com/datasets/subhajournal/android-malware-detection

[17] A. Parashar, A. Parashar, W. Ding, M. Shabaz, and I. Rida, "Data preprocessing and feature selection techniques in gait recognition: A comparative study of machine learning and deep learning approaches," Pattern Recognition Letters, vol. 172, pp. 65–73, Aug. 2023, doi: 10.1016/j.patrec.2023.05.021.

[18] I. Sumaiya Thaseen and C. Aswani Kumar, "Intrusion detection model using fusion of chi-square feature selection and multi class SVM," Journal of King Saud University - Computer and Information Sciences, vol. 29, no. 4, pp. 462–472, Oct. 2017, doi: 10.1016/j.jksuci.2015.12.004.

[19] L. Hu, L. Gao, Y. Li, P. Zhang, and W. Gao, "Feature-specific mutual information variation for multi-label feature selection," Information Sciences, vol. 593, pp. 449–471, May 2022, doi: 10.1016/j.ins.2022.02.024.

[20] A. Alhogail and R. A. Alharbi, "Effective ML-Based Android Malware Detection and Categorization," Electronics, vol. 14, no. 8, Art. no. 8, Jan. 2025, doi: 10.3390/electronics14081486.

[21] A. Muzaffar, H. Ragab Hassen, M. A. Lones, and H. Zantout, "An in-depth review of machine learning based Android malware detection," Computers & Security, vol. 121, p. 102833, Oct. 2022, doi: 10.1016/j.cose.2022.102833.

Analyzing Cyber Attack Detection in IoT Healthcare Environments Using Artificial Intelligence

Rawan Marzooq Alharbi, Muhammad Asif Khan
Department of Information Systems-College of Computer Science and Engineering, Taibah University, Madina, Saudi Arabia

***Abstract*—The rapid growth of the Internet of Things (IoT) has significantly increased its integration into daily life. In recent years, the integration of IoT technologies in healthcare has significantly enhanced patient care and operational efficiency. One of the most promising areas for using IoT devices in healthcare or interconnecting medical devices is known as the Internet of Medical Things (IoMT). IoMT supports various healthcare services, e.g., remote patient monitoring. However, there are serious cyber-security concerns, as various attacks have targeted these IoMT devices in recent years. This research presents an analytical approach to understanding how Artificial Intelligence (AI) can improve the detection of cyber-attacks within IoT healthcare environments. The main goal of this research is to provide an AI-based model to detect cyber-attacks in IoMT in the healthcare environment. Many researchers have worked on developing a framework in this field to address critical cybersecurity threats. However, these efforts often fall short of covering other important aspects such as data privacy and interoperability. In this study, a model and framework are proposed to monitor IoT networks, and detect potential security breaches in real-time to help in mitigating risks while maintaining healthcare services. The key findings contribute to strengthening cybersecurity protocols in healthcare IoT environments in order to ensure the protection of sensitive information against emerging cybersecurity vulnerabilities.**

Keywords*—*IoT healthcare security; cyber-attack detection; healthcare security; AI in healthcare; smart medical systems

I. Introduction

The Internet of Things (IoT) is a network of our daily life physical objects (things) that surround us and are connected together to send or receive data over internet without human intervention to ease human life. These things range from a wrist watch to home appliances and these are embedded with sensors which are smarter and more interactive with both us and the surrounding. The IoT promises to revolutionize our industries and economies by connecting billions of devices across the world. The IoT offers unparalleled opportunities to organizations for automation and novelty. As the technologies become established and matured, the IoT will be an integral part of industrial digital transformation. In future we will see spread of IoT from a smart highway to smart airports, smart home to smart cities, smart building to smart hospitals etc. There are various objects or things which have sensors such as car, door, lights, air conditioner etc. in different sectors can be connected through internet [1].

A survey conducted by Statista Inc [2] depicts there were 7.74 billion devices connected around the world in 2019 which could exceed 25 billion by 2030.

The exponential use of IoT devices in our daily lives has revolutionized various sectors including the healthcare sector. Thus, IoT is enabling the seamless connection of devices and systems to improve patient monitoring, diagnosis, and treatment [3]. Smart medical devices, and wearables have become integral components of modern healthcare, providing clinicians with real-time data and enhancing patient outcomes. Using IoT devices in healthcare or interconnecting medical devices is known as the Internet of Medical Things (IoMT) [4]. IoMT now supports various healthcare services, e.g., remote patient monitoring. However, this increased connectivity has also led to heightened vulnerabilities, making healthcare environments attractive targets for cyber-attacks [5].

The threats of a cyber-attack to target IoT healthcare systems mainly include data breaches, ransomware attacks, and denial-of-service (DoS) attacks [6]. The consequences of these attacks on IoT healthcare systems can be severe, ranging from compromised patient data to disruptions in critical healthcare services [7]. Since healthcare organizations strive to protect sensitive information and maintain operational integrity, traditional cybersecurity measures often fall short due to the complexity and scale of IoT networks. In this context, Artificial Intelligence (AI) emerges as a powerful tool for enhancing cybersecurity in IoMT environments. By leveraging advanced algorithms and machine learning techniques, AI can analyze vast amounts of data, identify patterns, and detect anomalies that may indicate potential threats.

With incessant growth in use and adoption of the IoT devices many IoT devices are found to have inherent security issues. The manufacturers of the devices focus on device functions and cost rather than the security features which in turn create security concerns and vulnerabilities in the devices. Wide spread of IoT devices and the information these devices need to provide features may be compromised, therefore, in this context cyber security has been extreme critical. A lot of research has tried to mimic a variety of hypothetical threats to enhance IoT security. Since more novel, and serious threats are being developed, requiring further work in this field especially in the healthcare sector. Moreover, the IoT devices only have limited amount of memory, computational capacity, and power. In such a situation, any communication issue caused by a cyber-attack might have hazardous consequences on a patient's health, and may even result in death.

Due to the rapid growth of infrastructure of IoT devices in recent years, IoT-based healthcare systems have become vital for detecting cyberattacks. Any attack may have a significant impact on the life of the patient. Therefore, the use of Artificial Intelligence (AI) is one of the finest solutions available to

improve the efficiency of detecting cyber-attacks in IoT healthcare environments. The aim of this research is to analyze the effectiveness of AI in detecting cyber-attacks within IoT healthcare environments, exploring both the current landscape and the future potential of AI-driven security solutions. In order to accomplish it, we have formulated the following research questions and we strive to address them in this research study:

RQ1: How integration of AI and Internet of Things (IoT) technologies can enhance cyberattacks detection and prevention capabilities in healthcare systems?

RQ2: What strategies can be employed to optimize AI cybersecurity model for scalable, cost effective and security compliant healthcare systems?

RQ3: How can AI improve the accuracy of cyberattack detection in IoT healthcare environment?

In order to find the answers of the research questions following hypotheses have been formed:

H_0 - AI Models improve the accuracy of cyberattack detection

H_1 - AI Model has the potential to enhance the detection of cyberattacks on IoT network in healthcare systems

The research study aims to develop an effective AI model to protect IoT healthcare systems by detecting cyberattacks in IoT healthcare networks and to provide an AI-based model that enhances the findings of existing research on an IoT healthcare security dataset. The proposed model aims to provide more accurate results from attack types compared to the relevant research of IoT healthcare datasets. The proposed model will get an optimum performance time compared to current strategies in relevant research of IoT healthcare datasets.

This research contributes to the body of knowledge by providing an efficient AI-based cybersecurity model for protecting IoT healthcare environment and for improving the accuracy and precision of cyber-attacks classification. Moreover, the model will show a potential to improve the sustainability of IoT systems by reducing the wastage of resources in identifying the cyber-attacks on time.

II. Literature Review

Due to the rapid advancement of technology, various communication and information technology-based solutions are being used in healthcare environments to facilitate both healthcare professionals and patients. These technologies mainly include the Internet of Things (IoT), artificial intelligence (AI), virtual reality (VR) and multi-agent systems. With the help of IoT technology, various gadgets having sensors and actuators are integrated to remotely monitor patients and provide different medical services like diagnosis, consultation, treatment, and information through on-ground healthcare staff [8]. Authors in study [9] proposed an IoT-based system to monitor the heartbeat and body temperature of a patient over a remote location through sensors and Bluemix technology. Similarly, authors in study [10] worked on a patient's activity recognition by analyzing the speech and movement patterns of a patient at home. They used a camera, breath sensor, and ECG for patient health monitoring. Likewise, researchers in study [11] developed a smartphone application to provide early medical assistance to elderly people at home.

With the progress of AI technology, healthcare data is now being analyzed and predicted through AI-based expert systems which help healthcare experts to make fast and efficient decisions for remote patients' healthcare. The AI models efficiently process the real-time sensor data and provide helpful information to healthcare professionals for better treatment decisions [12]. Similarly, VR technology is also used in telemedicine to improve the interaction between local healthcare staff and remote expert physicians for real-time patient healthcare including surgeries. Many recent research experiments have revealed that augmented reality (a type of VR) can improve the live motion-based interaction of remote healthcare experts for better diagnosis and treatment [13, 14].

Despite a lot of technological progress in healthcare environments, cybersecurity is a serious concern in IoT-based healthcare environments. Over the past few years, many researchers have worked on improving the security of IoMT-based healthcare systems. Authors in study [15] worked on detecting cyberattacks in IoT healthcare environments. The authors proposed an anomaly detection approach along with machine learning to detect cyberattacks in remote patient monitoring environments. The authors used the CICIoT dataset which contains 33 types of IoT attacks divided into seven main categories. The authors first pre-processed the CICIoT dataset for a balanced sample representation of each class. They further applied features- eliminating methods and features dimensionality reduction methods. Afterwards, they trained various machine learning models including Random Forest (RF), Logistic Regression (LR), Artificial Neural Network (ANN), and Adaptive Boosting (AB). Their proposed model achieved the highest accuracy based on Random Forest (RF), reaching up to 99.55%in binary and multiclass classification.

Similarly, researchers in [16] worked on detecting anomalies in IoT smart hospital environments. The authors proposed a low-latency anomaly detection system (ADS) and implemented it in the Contiki Cooja simulator. The authors achieved on average 80% accuracy in detecting three types of anomalies attacks including Rank and flooding attacks.

A study in [17] proposed an access control system for the privacy preservation of healthcare data in IoT healthcare architecture. The authors proposed a deep learning approach to secure IoT healthcare systems from unauthorized access by users and attackers. They also proposed a secure access control module by focusing on user attributes to protect IoT healthcare systems from cyberattacks caused due to unauthorized access. Their proposed model is based on a convolutional neural network (CNN) and it achieved 98% accuracy and 95% precision and F1-score.

In a study researchers [18] worked on improving the privacy and interoperability issues in IoT healthcare systems. The authors applied various data pre-processing techniques and trained a deep learning model that consists of a convolutional neural network (CNN) and Long Short-Term Memory (LSTM). Their proposed model achieved more than 80% accuracy, precision and recall on various datasets including PhysioNet, MIMIC, and eICU datasets.

In another study [19] proposed the cognitive machine learning based Cyber-Physical healthcare attack detection model to transmit healthcare data mules firmly. The collected data stored over the cloud platform where machine learning algos analyze the cyberattack patterns and predict their behavior accordingly. The proposed Cyber Machine Learning Anomaly Detection Framework (CML-ADF), can detect the cyberattacks over the edge nodes at the physical layer. The experimental results depict, CML-ADF framework achieving an accuracy of 98.2 % with an efficiency ratio of 97.8 % respectively. Moreover, it also outburst the delay and communication cost parameter over existing models with the ratio of 21.3 % and 18.9 % respectively.

The researchers in [20] demonstrate the Intrusion Detection System (IDS) under the umbrella of Internet of Medical Things (IoMT) based on Deep Recurrent Neural Network (DRNN) and supervised Machine Learning (SML) classifiers to forecast the unknown cyberattacks. In order to optimize features, the authors used the bio-inspired particle swarm approach which determines the outstanding results since its inception in the real-world optimization challenges. Under rigorous test beds and experimentation, the proposed SML model outperformed the existing approaches while attaining the accuracy of 99.76 %.

In a study, researchers [21] worked on the AI assisted IoT-CPS framework to diagnose patient diseases such as Diabetes, Heart anomalies and Gait disorders. Each disease has its own set of detection parameters with respect to patient gender, age and complexity. The author used the openly available dataset, acquired from Kaggle repository for the execution of proposed AI-enabled IoT-CPS methodology. The experimental tests showed that the proposed framework outperformed the existing studies with an accuracy of 86.4 % indeed.

A research study [22] proposed the framework to generate IoT based normal and malicious data in order to develop the IoT assisted context aware security solutions for malicious traffic detection in various use cases. The author named the proposed open-source IoT data generator as IoT-Flock. It not only generates the traffic but also converts the captured data into a particular dataset for further analysis and recommendation. While using the proposed framework, the author generates the IoT-healthcare dataset which comprises both normal and IoT attack affected data which is ultimately being detected with the help of multiple ML based classifieds to protect the healthcare system from cyberattacks. The proposed context aware IoT based data generator is very effective to analyze the sensory IoT healthcare use cases in terms of cyber security.

The authors in [23] developed the cyber-attack inclusion with anomaly detection methodology accompanying recursive feature elimination (RFE) and multi-layer perceptron (MLP) approach. Both of these approaches are very effective to identify optimal feature selection and performance evaluators. The proposed model was tested on various IoMT cyber security datasets with an average accuracy of 98.08 % respectively.

In a study [24] proposed the AI assisted Artificial Fish Swarm-driven Weight normalized Adaboost (AF-WAdaBoost) model for the optimization of accuracy and sustainability parameters to identify cyberattacks in IoT healthcare systems. The proposed AF-WAdaBoost mechanism fiercely adjusts the ML classifiers to enhance accuracy and persistence against evolving threats. The experimental study shows the proposed model outperformed traditional IoT- healthcare cyberattack detection approaches with an accuracy of 98.69 %, F-measure and Precision with 94.86 % and 95.72 % respectively.

Researchers in study [25] worked on the novel cyberattack healthcare anomaly detection technique based on Cyber Physical System (CPS) to enhance network security. A quantum cloud Federated learning mechanism has been implemented on the CPS in order to build an IoT healthcare network. Existing approaches lack to achieve reasonable values for Round Trip Time (RTT) using Transmission control Protocol (TCP) packets and are ineffective on the large number of packets. Our proposed model attained network efficiency of 92 %, security analysis with 89 %, training and validation loss of 79 % and 49 % respectively.

A research study [26] suggested the novel approach for assessing the cybersecurity of e-healthcare applications i.e., assistance of quantum machine learning. In this approach the author proposed the deep variational adversarial network encoder with fuzzy Gaussian quantile neutral network to identify and classify the useful characteristics of user activity data patterns to identify the vicious user to enhance the overall network security. The implementation of the proposed approach shows vigorous results in terms of learning rate, prescient misfortune, transmission influence, jitter, throughput and dynamic serverless response time. Moreover, the proposed system attained the random accuracy of 98%, F-1 score of 75 %, Mean Average Precision of 65 % and the kappa Coefficient of 69% which outperformed the existing server based E-healthcare security approaches.

In a study [27] illustrated the comparison between various AI-driven threat detection approaches to enhance security and to mitigate cyber threats in the IoT Healthcare environment. In the world of AI-driven cyber security solutions, the most prominent are transformer based models, federated learning and blockchain integration to step into the real-time threat detection systems. The most prominent approach to mitigate cyberattack in the Enhanced AI-based Network intrusion Detection using Generative Adversarial Networks (GANs) which attained the accuracy of 98.2 % which exceeded the preceding resilient intrusion detection system and multi-domain trojan detection framework which were at 97.8 % and 97.1 % respectively.

The authors in [28] illustrated how future healthcare hubs and proactive smart devices implement their own cybersecurity models to ensure full proof AI assisted healthcare communication channels. The proposed AI based healthcare communication cyber security model set the digital healthcare communication IEEE standards interviewing the cybersecurity policies. Moreover, the Joint Optimized Infrastructure for Network Empowered Research (JOINER) came into existence to ensure secure future communication in the healthcare sector. Now each AI-driven emerging cybersecurity policy relies on JOINER architecture. The proposed approach outperformed the existing communication trends, in terms of data rate, mobility, latency, connectivity gap, and reliability.

A study [29] introduced the framework of trustworthiness and decision-making support mechanism within the internet of

medical things (IoMT) use cases. Keeping in view the shortcomings of risk management and management approaches to accessed the IoMT context aware scenarios in the existing models, lack proper automation and inability to mitigate the various security risks in the healthcare system, the proposed (MLRA-Sec) framework integrate the Hybrid Risk Assessment (RA) model with ML-based anomaly detection technique to ensure and evaluate cumulative IoMT risk. Experimental study shows effective outcome as compared to the state of the art intrusions detection IoT-ML based models.

A research study conducted by study [30] focused on addressing the problem of cyber risk assessment in IoT healthcare environments. The authors proposed a lightweight, efficient and dynamic approach for risk assessment of security events in IoT healthcare infrastructure. The authors generated synthetic data and then executed multiple simulation scenarios on it while mapping the attack surface and applying threat models. The proposed approach not only highlight the security risks but also provides some helpful information to mitigate the cyber threats.

Authors in study [31] highlighted the challenges in the healthcare security system with the advancement of energy constraint communication devices. Considering the severity of the issue, the author developed the monitoring frequency-based detection and dynamic threshold mitigation method using Temporal Convolutional Networks (TCNs) in the 5G healthcare IoT (H-IoT) environment. The proposed approach calculates the H-IoT node's incoming and outgoing data mules counts for five seconds with respect to overall data traffic. This dynamic approach provides adaptive security by taking the mean value of detected malicious data across all nodes to evaluate the threshold cap to enhance the classification accuracy and restricts the true DDos attacks. The experimental test beds conducted under two communication protocols i.e., Message Queuing Telemetry Transport (MQTT) and User Datagram Protocol (UDP) in a realistic 5G based healthcare environment. The proposed model attained the average accuracy of 99 % on MQTT datasets and 99.99 % on UDP dataset with 80 % mitigation.

A research work carried out by study [32] showed the impact and comparison of AI-driven existing IoMT security models. With recent advancement in IoMT, in spite of its effectiveness, it also becomes the easy corner for hackers to breach the patient's sensitive information and use it for their own evil desires. To resiliate these attacks, novel taxonomy of AI based intrusions detection IoMT schemes have been highlighted in this article. Moreover, tasks and processes are carried out over Cloud-Fog-Edge architectures to reduce the overall latency and efficient use of computational resources in real-time to enhance the efficiency of the system, also being a part of its comparison approaches.

Authors in study [33] focused on the vulnerabilities to cyberattacks in the IoT healthcare system. Along with the rapid use of heterogeneous smart devices becomes a more severe concern as the time proceeds. In this paper, author proposed the deep neural network assisted cyberattack detection model to accompanying the unknown cyberattacks in IoT healthcare environment. An AI based proposed anomaly cyber threats have been tested on latest ECU-IoHT dataset and attained the accuracy of 99.85% under the average area receiver characteristic cure is at 0.99 with the false positive is 0.01 that highlights the effectiveness of the proposed cyberattack detection system as compared to the state of the art approach.

Researchers in study [34] worked on preventing cyber threats in IoT healthcare environments. The authors proposed a framework to detect different types of cyberattacks in IoT healthcare networks. The authors used four different types of dataset related to cybersecurity attacks for having maximum cyberattacks coverage and getting a more generalized AI model. The authors applied different data pre-processing techniques like data cleaning, data imputation, and dimensionality reduction. Afterwards, they trained the pre-processed data over an attention-based bi-LSTM deep convolved network to detect the cyber-attacks in IoT healthcare environments. Overall, the authors achieved around 99% accuracy on all datasets.

A research study [35] proposed a hybrid approach to detect cyber-attacks in internet of medical things (IoMT) environment. In order to efficiently detect the cyber-attacks, the authors integrated three AI models which include K-nearest neighbors (KNN), long-short term memory (LSTM), and principal component analysis (PCA). Both the LSTM and PCA models were used for data pre-processing. Afterwards, they trained and tested KNN model for detect cyber-attacks in IoMT environment. The authors tested the proposed hybrid framework over four datasets which include TON-IoT, ECU-IoHT, ICU, and WUSTL-EHMS dataset. Overall, the proposed approach resulted 99% accuracy for detecting cyber-attacks in IoMT environment.

Similarly, researchers in study [36] worked on efficiently detecting the intelligent detection of intrusions in IoMT environment. The authors integrated two deep learning models which include convolutional neural network (CNN) and LSTM for efficient detection of intrusions. The CNN was mainly used for features extraction whereas the LSTM was used sequential network traffic flows prediction. The authors referred the proposed intrusion detection system (IDS) as hybrid deep learning-based IDS solution for IoMT (HIDS-IoMT). The proposed approach was also tested in a fog computing environment where the proposed HIDS-IoMT model was deployed on a Raspberry Pi device which not only demonstrated an accurate detection of intrusions but also reduced the detection latency as the IDS was deployed on an edge device.

Likewise, authors in study [37] designed a framework for intrusion detection in IoMT environments. The authors proposed a stacking ensemble approach by stacking machine learning and deep learning models into two groups to effectively secure the IoMT network. The authors used ECU-IoHT dataset. They first pre-processed the dataset by applying data cleaning, data normalization, data balancing and features selection techniques. The proposed framework combined both machine learning and deep learning classifiers to detect various cyber-attacks in IoMT environment. Moreover, in order to deploy the proposed AI-based IDS in real-time [38], the authors implemented Kappa architecture which helped to update data streams with low latency. The proposed approach resulted 99.1%accuracy for binary class classification and 99.3%accuracy for multi-class classification to detect four types

of cyber-attacks including spoofing, denial of service, scanning, and Smurf attack.

Authors in study [39] proposed a hybrid deep learning framework to enhance the security and quality of service (QoS) in IoT healthcare systems. The authors introduced a novel software-defined network (SDN) based architecture and integrated it with a security module that is responsible to detect cyberattacks in healthcare network. The secure module was mainly developed by training and testing a Bidirectional LSTM model over CICDDoS2019 dataset. In order to improve the QoS by reducing transmission, the authors used CNN model in SDN controller to find the best fog node integrated with SDN which further improved the overall network life. The proposed hybrid deep learning framework resulted 99.59% accuracy, 99.53% F1-score, and 3 ms delay in order to detect cyber-attacks in IoT healthcare environments.

The literature reviewed above provided us information of some datasets used to detect, train and test AI model in healthcare environment. Table I shows short description of these datasets:

TABLE I. DATASETS USED TO DETECT, TRAIN AND TEST AI MODEL IN HEALTHCARE ENVIRONMENT

Dataset	Reference	Description
CICIoT	Khan & Alkhathami, 2024	This dataset consists of real-time normal and cyberattacks traffic. It is widely used to train and test AI models to efficiently detect cyberattacks. It includes real-time network traffic of 105 real IoT devices. It contains mainly 156 features and covers 33 types of cyber-attacks data
Aegean AWID	Pimple & Sharma, 2025	This dataset contains the network attacks traffic of Wi-Fi attacks. It has 156 features
IoT-ICU	Hussain et al., 2021	This dataset mainly focused on ICU environment where different sensors and actuators are connected over a Wi-Fi. It contains different types of cyberattacks and normal traffic
NSLKDD	Saheed & Arowolo, 2021	It is an updated version of KDD intrusion detection dataset. It has some class imbalance issues
ECU-IoHT	Kilincer et al., 2023	This dataset was mainly developed to analyze the vulnerabilities in IoT healthcare environment
CICIDS2017, UNSW-NB15	Alsulami, 2024	These are basically general cyberattack dataset. But these are also used in healthcare environment
IoT-23, LINET 2020, NetML	Prabakar et al., 2024	This dataset mainly contains IoT malware related data to detect and figure out the potential attacks in healthcare environments
Bot-IoT	Ksibi et al., 2025	This dataset mainly contains botnet attack traffic

III. RESEARCH METHODOLOGY

In order to conduct research for finding out solution of the research questions stated earlier, we adopted design science research methodology which consists of five major stages. The Fig. 1 shows the five stages including data collection, data preprocessing, feature selection, AI models training and performance comparison.

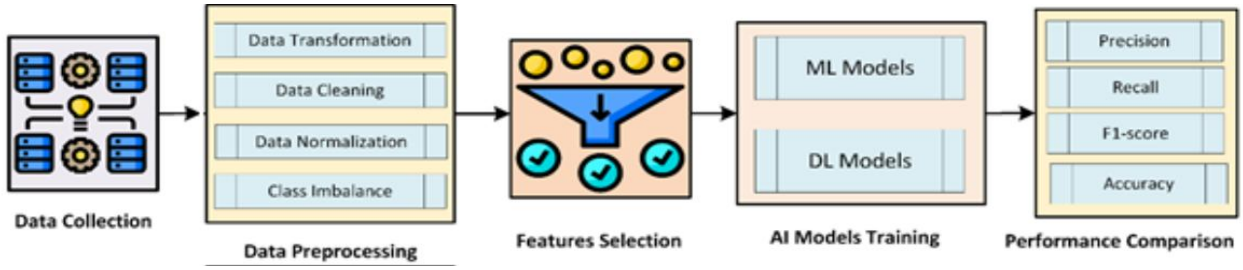


Fig. 1. The proposed methodology for cyber-attacks detection using AI in IoT healthcare environments.

A. Data Collection

The data collection is a primary step of our proposed methodology. To go through further steps to recognize cyber-attack detection using AI in IoT healthcare environments, we first need to collect a dataset. A few researchers have worked on detecting cyber-attacks in IoT healthcare environments. Therefore, we reviewed existing datasets and then selected the dataset with a vast variety of cyber-attacks. The IoT healthcare security dataset named as CIC IoMT dataset (Dadkhah et al., 2024) contains eighteen (18) types of cyber-attacks traffic including spoofing, scanning, TCP, UDP, and MQTT attacks. These eighteen (18) types of cyber-attacks are categorized into five classes which include spoofing, reconnaissance (RECON), MQTT, denial of service (DoS) and distributed DoS (DDoS) attacks. The complete mapping of eighteen (18) types of cyber-attacks along with their classified category are shown in Table II.

TABLE II. CYBERATTACKS WITH CLASSIFIED CATEGORIES

Category	Cyber-attacks
SPOOFIN	ARP Spoofing
RECON	Ping Sweep, Recon VulScan, OS Scan, Port Scan
MQTT	Malformed Data, DoS Connect Flood, DDoS Publish Flood, DOS Publish Flood, DDoS Connect Flood
DOS	DOS TCP, DOS ICMP, DOS SYN, DOS UDP
DDOS	DDoS SYN, DDOS TCP, DDOS ICMP, DDOS UDP

The CIC IoMT healthcare security dataset (Dadkhah et al., 2024) is mainly saved in packet capture (.pcap) file format collected via WireShark tool. A few .pcap files are also converted into the .csv file format. The CIC IoMT dataset (Dadkhah et al., 2024) contains network traffic of 40 IoMT devices including 25 real devices and 15 simulated devices. The dataset contains three types of communication protocols used by

healthcare devices which include Wi-Fi, Bluetooth, and MQTT devices. The MQTT-based healthcare devices are simulated devices whereas the Bluetooth and Wi-Fi-based healthcare devices are real devices integrated into a real-time network. The CIC IoMT dataset (Dadkhah et al., 2024) has 45 features in each .csv file without labels. These features are enlisted and briefly described in Table III. In summary, the CIC IoMT dataset (Dadkhah et al., 2024) is big dataset contains multiple .pcap and .csv files and each .csv file contains samples in range 5000 to 80,000 samples. There is a minor issue of class imbalance if look at it from the perspective of binary or multi-class classification.

TABLE III. LIST OF FEATURES WITH BRIEF DESCRIPTION IN CIC IOMT DATASET (DADKHAH ET AL., 2024)

Feature	Description
Header Length	Mean of the header lengths of the transport layer
Time-To-Live	Time-to-live
Rate	Speed of packet transmission within a window in packets/sec
fin flag number	Proportion of packets with FIN flags in the window
syn flag number	Proportion of packets with SYN flags in the window
rst flag number	Proportion of packets with RST flags in the window
psh flag number	Proportion of packets with PSH flags in the window
ack flag number	Proportion of packets with ACK flags in the window
ece flag number	Proportion of packets with ECE flags in the window
cwr flag number	Proportion of packets with CWR flags in the window
syn count	Count of Syn flag occurrences in packets
ack count	Count of Ack flag occurrences in packets
fin count	Count of Fin flag occurrences in packets
rst count	Count of Rst flag occurrences in packets
IGMP	Average number of IGMP packets in the window
HTTPS	Average number of HTTPS packets in the window
HTTP	Average number of HTTP packets in the window
Telnet	Average number of Telnet packets in the window
DNS	Average number of DNS packets in the window
SMTP	Average number of SMTP packets in the window
SSH	Average number of SSH packets in the window
IRC	Average number of IRC packets in the window
TCP	Average number of TCP packets in the window
UDP	Average number of UDP packets in the window
DHCP	Average number of DHCP packets in the window
ARP	Average number of ARP packets in the window
ICMP	Average number of ICMP packets in the window
IPv	Average number of IPv packets in the window
LLC	Average number of LLC packets in the window
Tot Sum	Total packet length within the aggregated packets (window)
Min	Shortest packet length within the aggregated packets (window)
Max	Longest packet length within the aggregated packets (window)
AVG	Mean packet length within the aggregated packets (window)
Std	Standard deviation of the packet length within the aggregated packets (window)
Tot Size	(Avg.) Length of the packet
IAT	Interval mean between the current and previous packet in the window
Number	Total number of packets in the window
Variance	Variance of the packet lengths in the window
Protocol Type	Mode of protocols in the window

B. Data Preprocessing

Once we are done with data collection, the next step is to pre-process the dataset in order to pass a refined dataset to AI models for better training. As shown in Fig. 2, the first step in data pre-processing is data transformation, i.e., to transform the categorical values to numeric integer values because the AI models only work on number values. So, to perform data transformation, we used the one-hot encoding technique After the data transformation, we performed data cleaning to remove missing values, null values or infinite values from the dataset. Once the dataset is cleaned, we then need to normalize the dataset to scale the feature values in a comparable range. For this purpose, we used Min-Max normalization which scales the feature values in the range [0, 1].

We noticed that there was a class imbalance issue in the dataset. To fix the class imbalance issue, we used the down-sampling technique as there were some attacks where samples were less than 5000. So, when we combined the data-frames of such attack samples based on same class, we came to know that minimum 18000 samples were present in a class. Therefore, to fix the class imbalance issue, we dropped down the samples from other attack classes by randomly selecting 18000 samples from each class via down-sampling technique. Eventually, after the down-sampling, we had 18000 samples from each of five attack classes as mentioned in Table II which in total leads to 18,000 x 5 = 90,000 attack samples. Similarly, we randomly selected 18000 samples from benign, i.e., normal class samples as well. So in total, we had 90,000 (attack samples of five classes) + 18000 (benign samples of 1 class) = 108,000 samples for training and testing AI models.

C. Features Selection

Although the dataset is pre-processed, we further need to perform features selection to provide the most useful features to machine learning models for better performance. For this purpose, we applied four commonly used features selection techniques, i.e., Anova, Chi2, mutual information, and extra tree features selection method to select only useful features from the dataset. Using these four features selection techniques, we tried different numbers of features i.e., 20, 30 features out of total 45 features and check on which feature set the machine learning models yield better performance by comparing the results with the performance achieved while training AI models on all features. Therefore, we considered three scenarios.

- Training and testing AI models on all features.
- Training and testing AI models on 20 features selected via features selection methods.

- Training and testing AI models on 30 features selected via features selection method.

D. Models Training

Before starting training to AI models, we first need to divide the dataset into a training set and a testing set. To split the dataset, we use the 80:20 ratio, i.e., 80% dataset were split randomly into training-set and 20% data were used for testing-set. As shown in Fig. 2, we trained both machine learning models and deep learning models for cyber-attacks detection in IoT healthcare environments. Therefore, we trained five commonly used machine learning models, i.e., decision tree (DT), random forest (RF), Naïve Bayes (NB), k-nearest neighbors (KNN), and logistic regression (LR). Similarly, we trained two commonly used deep learning models, i.e., multi-layer perceptron (MLP), and convolutional neural network combined with long-short term memory (CNN+LSTM) model, for cyber-attacks detection in IoT healthcare environments.

E. Performance Comparison

We compared the performance of all trained models to decide the best-performing model. We used four commonly used performance metrics including accuracy, precision, recall and F1-score. These evaluation metrics are defined in the next results and discussion section.

IV. Results

In order to conduct experiment, we acquired publicly available dataset and proceeded with further stages. For the data preprocessing, data transformation and training and testing of the AI models for cyber-attacks detection in IoT healthcare environments, we used Python programming language and some standard AI models and libraries, such as numpy, pandas, scikit-learn, and tensorflow.

To begin with experimentation, we considered three scenarios in which seven AI models are trained for cyber-attacks detection in IoT healthcare environments. These seven commonly used AI models include decision tree (DT), random forest (RF), Naïve Bayes (NB), k-nearest neighbors (KNN), logistic regression (LR), multi-layer perceptron (MLP), and convolutional neural network combined with long-short term memory (CNN_LSTM) model. All these models were trained with their default parameters set in Python's scikit-learn library except MLP and CNN_LSTM. For MLP and CNN_LSTM, we used tensorflow library. We further used Adam optimizer, set learning rate = 0.001, defined four hidden layers. After training these models, we then need to evaluate their performance on test data for cyber-attacks detection. For this purpose, we used four commonly used performance evaluation parameters, i.e., precision, recall, F1-score, and accuracy. These performance matrices are defined as follows:

1) Precision (PR): It tells us about the capacity of a system to accurately recognize an attack upon the happening of an actual cyber-attack. It presents a correlation between accurately anticipated attacks and actual outcomes. Mathematically, precision is expressed as:

$$PR = \frac{True\ Positives}{\text{True Positives } + \text{ False Positives}} \times 100$$

2) Recall (RE): It tells us about the correctly identified cyber-attack events upon its existence in the network. Mathematically, recall is written as:

$$RE = \frac{True\ Positives}{\text{True Positives } + \text{ False Negatives}} \times 100$$

3) Accuracy (AC): It tells us about how many attack events are categorized as attacks and how many regular network packets are categorized as normal traffic. It also indicates the proportion of accurate predictions relative to the total number of samples. Mathematically, it is expressed as:

$$AC = \frac{\text{True Positives } + \text{ True Negatives}}{\text{Total Samples}} \times 100$$

*4) F1- score:*It is defined as a harmonized average of recall and precision. It tells us about the percentage of correctly predicted testing samples as cyber-attacks. Mathematically, it is expressed as:

$$F1\ score = 2 \times \frac{PR\ x\ RE}{PR + RE}$$

In order to evaluate the best performance among the AI models, we have three scenarios as we mentioned earlier and will find out the best-performing model to recognize cyber-attacks in each scenario.

B. Scenario 1 - Training and Testing AI Models on All Dataset Features

We applied the data preprocessing techniques stated earlier and after cleaning and normalization of the data, we selected all features to test and compare performance of the seven AI models. Table IV shows overall performance of seven AI models for cyberattacks detection in IoT healthcare networks over the test data.

TABLE IV. Performance of All Seven AI Models when All Features Selected

Classifier	Accuracy	Precision	Recall	F1 score
DT	0.97	0.9	0.97	0.97
RF	0.98	0.98	0.98	0.98
NB	0.45	0.41	0.46	0.35
KNN	0.94	0.94	0.94	0.94
LR	0.36	0.40	0.36	0.32
MLP	0.52	0.56	0.52	0.49
CNN_LSTM	0.84	0.85	0.84	0.84

It can be observed that RF classifier efficiently identified critical cyberattacks whereas the LR classifier poorly performed. Overall, the RF classifier resulted 98% accuracy, 98% precision, 98% recall and 98% f1 score for cyberattacks detection IoT healthcare environments. Fig. 2 shows the performance of all the seven models when all features were selected.

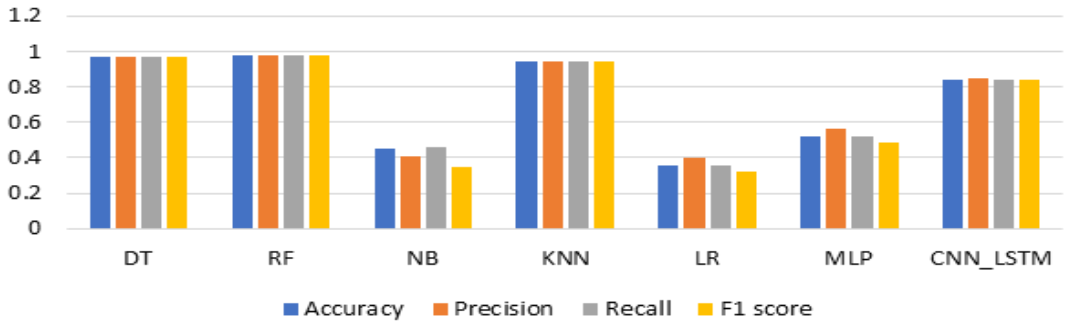


Fig. 2. Test results of all seven AI models when all features selected.

C. *Scenario 2 - Training and Testing AI Models on Top 20 Features*

After the data preprocessing, we selected top 20 features to test and compare the performance of AI models on all features vs. a subset of features according to proposed methodology for cyberattacks detection in IoT healthcare environments. We applied four features selection techniques including Anova, Chi^2, Mutual information and Extra tree using scikit-learn library in Python programming language. Table V shows the accuracy and recall scores of seven AI models for cyberattacks detection in IoT healthcare networks over the test data.

It can be observed that for Accuracy, the RF classifier efficiently identified critical cyberattacks whereas the LR classifier poorly identified cyber-attacks. Overall, the RF classifier resulted 98% accuracy for Anova, 97% accuracy for Chi2, 97% accuracy for Mutual information and 97% accuracy for Extra Tree features selection technique for cyberattacks detection IoT healthcare environments. Similarly, for Recall, the RF classifier efficiently identified critical cyberattacks whereas the LR classifier poorly performed. Overall, the RF classifier resulted 89% recall for Anova, 98% recall for Chi^2, 98% recall for Mutual information and 98% accuracy for Extra Tree features selection technique for cyberattacks detection IoT healthcare environments. Fig. 3 shows accuracy and recall of the seven AI models on top 20 selected features.

We obtained data of Precision and F1 scores for all AI models when 20 features were selected. Table VI shows the data.

TABLE V. ACCURACY AND RECALL SCORES OF SEVEN AI MODELS WHEN 20 FEATURES SELECTED

Classifier	Accuracy				Recall			
	Anova	Chi2	Mutual_Info	Extra_Tree	Anova	Chi2	Mutual_Info	Extra_Tree
DT	0.86	0.97	0.96	0.96	0.87	0.97	0.97	0.97
RF	0.98	0.97	0.97	0.97	0.89	0.98	0.98	0.98
NB	0.37	0.45	0.45	0.45	0.38	0.46	0.46	0.45
KNN	0.78	0.94	0.94	0.94	0.78	0.94	0.94	0.95
LR	0.40	0.35	0.35	0.29	0.41	0.35	0.35	0.29
MLP	0.70	0.46	0.48	0.42	0.71	0.47	0.49	0.43
CNN_LSTM	0.84	0.83	0.84	0.84	0.84	0.85	0.84	0.84

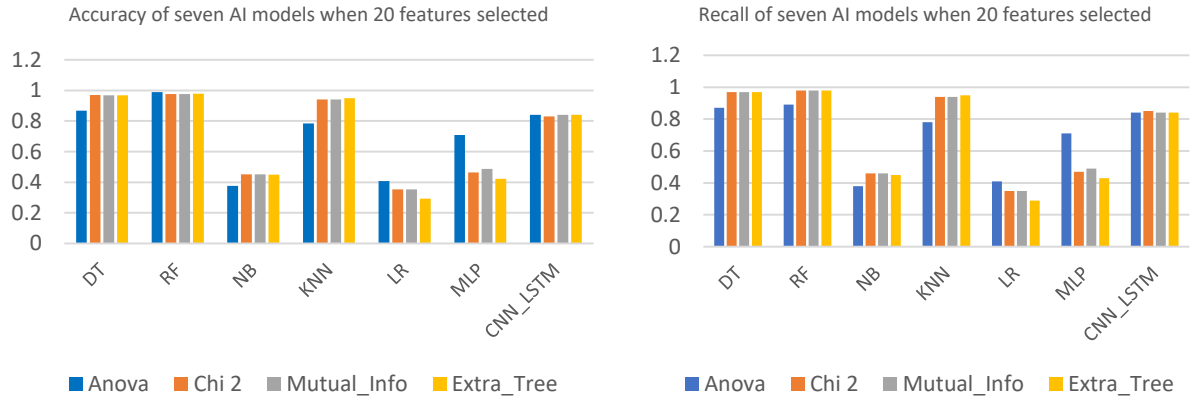


Fig. 3. Test results of Accuracy and Recall of seven AI models when 20 features selected.

TABLE VI. PRECISION AND F1 SCORES OF SEVEN AI MODELS WHEN 20 FEATURES SELECTED

Classifier	Precision				F1 score			
	Anova	Chi2	Mutual_Info	Extra_Tree	Anova	Chi2	Mutual_Info	Extra_Tree
DT	0.47	0.97	0.97	0.97	0.87	0.97	0.97	0.97
RF	0.89	0.98	0.98	0.98	0.89	0.98	0.98	0.98
NB	0.38	0.41	0.41	0.42	0.33	0.36	0.36	0.36
KNN	0.79	0.94	0.94	0.95	0.78	0.94	0.94	0.95
LR	0.26	0.39	0.39	0.29	0.30	0.29	0.29	0.25
MLP	0.77	0.64	0.53	0.44	0.70	0.43	0.46	0.37
CNN_LSTM	0.84	0.85	0.84	0.84	0.84	0.85	0.84	0.84

It can be observed that the RF classifier efficiently identified critical cyber-attacks whereas the LR classifier poorly identified cyber-attacks. Overall, the RF classifier resulted 89% precision for Anova, 98% precision for Chi^2, 98% precision for Mutual information and 98% precision for Extra Tree features selection technique for cyber-attacks detection IoT healthcare environments. Similarly, F1 scores show that RF classifier efficiently identified critical cyber-attacks whereas the LR classifier poorly identified cyber-attacks. Overall, the RF classifier resulted 89% f1-score for Anova, 98% f1-score for Chi^2, 98% f1-score for Mutual information and 98% f1-score for Extra Tree features selection technique for cyber-attacks detection IoT healthcare environments. Fig. 4 shows Precision and F1 scores of the seven AI models on top 20 selected features.

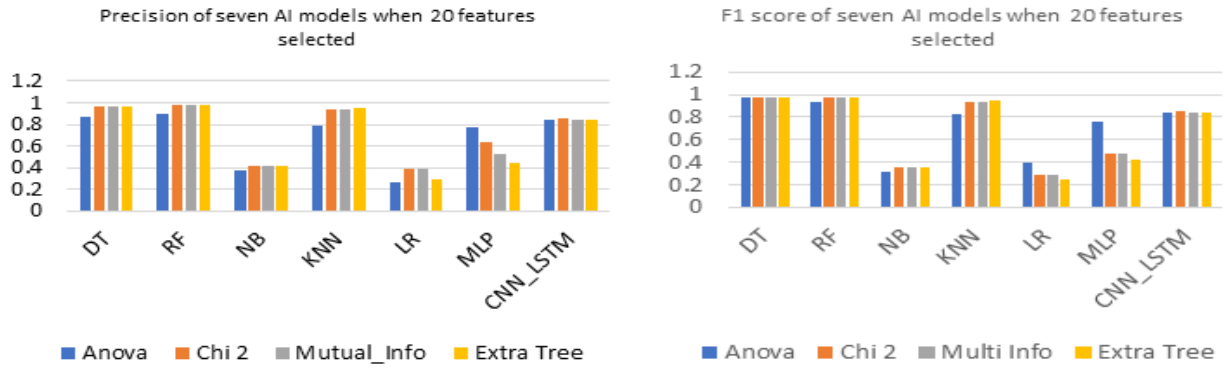


Fig. 4. Test results of Precision and F1 scores of seven AI models when 20 features selected.

D. Scenario 3 - Training and Testing AI Models on Top 30 Features

In this scenario, we selected top 30 features to test and compare the performance of AI models on all features vs. a subset of features according to the proposed methodology for cyber-attacks detection in IoT healthcare environments. Table VII shows the accuracy and recall scores of seven AI models for cyberattacks detection in IoT healthcare networks over the test data.

It can be observed that RF classifier efficiently identified critical cyberattacks whereas the LR classifier poorly identified the cyberattacks. Overall RF resulted 93% accuracy and recall for Anova, 98% accuracy and recall for Chi^2, 98% accuracy and recall for Mutual_Info and 98% accuracy and recall for Extra_Tree features selection techniques for cyberattacks detection in IoT healthcare environments. Fig. 5 shows the Accuracy and Recall scores metrics of seven AI models for 30 features selected.

TABLE VII. ACCURACY AND RECALL SCORES OF SEVEN AI MODELS WHEN 30 FEATURES SELECTED

Classifier	Accuracy				Recall			
	Anova	Chi2	Mutual_Info	Extra_Tree	Anova	Chi2	Mutual_Info	Extra_Tree
DT	0.91	0.97	0.97	0.97	0.91	0.97	0.97	0.97
RF	0.93	0.98	0.98	0.98	0.93	0.98	0.98	0.98
NB	0.38	0.45	0.45	0.45	0.38	0.46	0.46	0.45
KNN	0.82	0.94	0.94	0.95	0.82	0.94	0.94	0.95
LR	0.44	0.35	0.35	0.30	0.44	0.35	0.35	0.29
MLP	0.74	0.5	0.5	0.46	0.74	0.5	0.5	0.46
CNN_LSTM	0.84	0.85	0.84	0.84	0.84	0.85	0.84	0.84



Fig. 5. Test results of Accuracy and Recall scores of seven AI models for 30 features selected.

We obtained the data of Precision and F1 scores of seven AI models for cyberattacks detection in IoT healthcare networks over the test data. Table VIII shows the data.

The RF classifier efficiently identified critical cyberattacks whereas LR poorly performed in identifying the cyberattacks. Overall RF resulted 93% precision and F1 for Anova, 98% precision and F1 for Chi^2, 98% precision and F1 for Mutual_Info and 98% precision and F1 for Extra_Tree features selection techniques for cyberattacks detection in IoT healthcare environments. Fig. 6 shows the Precision and F1 scores metrics of seven AI models for 30 features selected.

TABLE VIII. ACCURACY AND F1 SCORES OF SEVEN AI MODELS WHEN 30 FEATURES SELECTED

Classifier	Precision				F1 scores			
	Anova	Chi2	Mutual_Info	Extra_Tree	Anova	Chi2	Mutual_Info	Extra_Tree
DT	0.91	0.97	0.97	0.97	0.91	0.97	0.97	0.97
RF	0.93	0.98	0.98	0.98	0.93	0.98	0.98	0.98
NB	0.37	-0.41	0.41	0.42	0.31	0.36	0.36	0.36
KNN	0.82	0.94	0.94	0.95	0.82	0.94	0.94	0.95
LR	0.45	0.39	0.39	0.29	0.39	0.29	0.29	0.25
MLP	0.86	0.58	0.58	0.67	0.76	0.47	0.47	0.42
CNN_LSTM	0.84	0.85	0.84	0.84	0.84	0.85	0.84	0.84

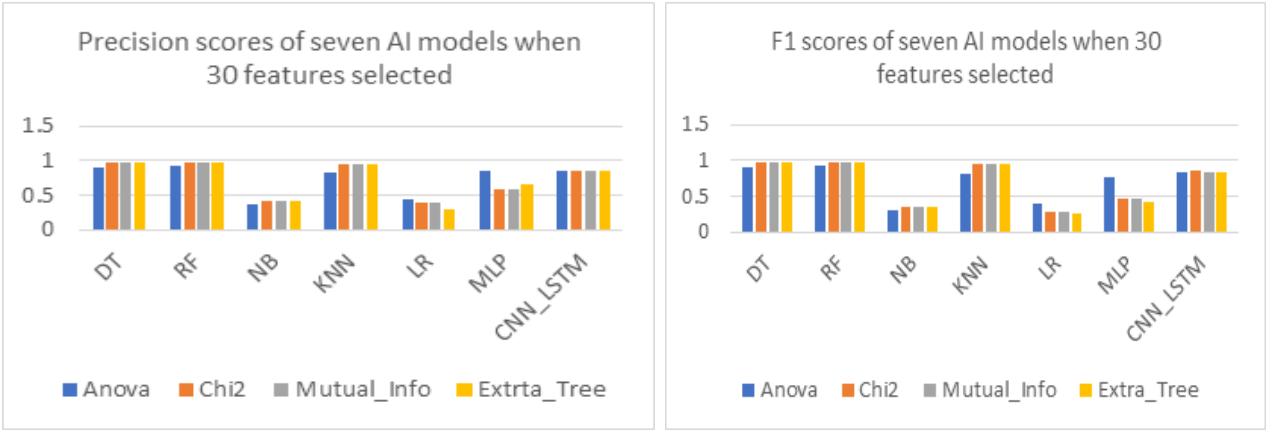


Fig. 6. Test results of Precision and F1 scores of seven AI models for 30 features selected.

V. DISCUSSION

The overall results in all three scenarios revealed that the RF classifier outperformed all other AI models for cyber-attacks detection in IoT healthcare environments. The RF classifier resulted the highest scores in terms accuracy, precision, recall, and F1-score for all three scenarios and across all four features selection techniques. In contrary, the LR classifier resulted the lowest scores in terms accuracy, precision, recall, and F1-score for all three scenarios and across all four features selection techniques. The experimental results support that both our hypothesis H0 and H1 are true since it is proved that the AI models improve the accuracy of cyber-attacks detection in IoT healthcare environments which eventually leads to prove that AI models has potential to enhance the detection of cyber-attacks in IoT healthcare environments.

VI. CONCLUSION AND FUTURE WORK

With the rapid application of IoT devices in healthcare environments, the cyber-attacks are surging in IoT healthcare environment due to limited security features such as limited

memory and processing power integrated in IoT devices. The healthcare environment is highly sensitive and any malfunction of IoT healthcare devices due to cyber-attacks can lead to severe implications even death. Therefore, it is highly essential to secure IoT healthcare devices from cyber-attacks. Hence, in this research, we proposed a methodology for cyber-attacks detection in IoT healthcare environments using AI. The proposed methodology consists of five major stages including data collection, data pre-processing, features selection, AI models training and performance comparison. We acquired a publicly available CIC IoMT dataset for training & testing of the AI models for cyber-attacks detection in IoT healthcare environments using Python language and some standard AI models and libraries. We used four commonly used features selection methods and trained and tested five commonly used machine learning models and two deep learning models. Furthermore, we considered three scenarios in which all seven AI models are trained for cyber-attacks detection in IoT healthcare environments. The experimental results revealed that the Random Forest (RF) classifier outperformed all other AI models for detecting 18 types of cyber-attacks—mapped into five attack classes—and normal traffic in IoT healthcare environments having 98% accuracy, 98% precision, 98% recall and 98% F1-score. Likewise, the best results were achieved on selecting 20 features out of 45 features using mutual information features selection method.

In future, we aim to increase the attack coverage to enable AI model to detect more types of cyber-attack. Furthermore, we aim to deploy the proposed solution in real-time environment in order to test its performance in real-time use case.

REFERENCES

[1] J. Deogirikar and A. Vidhate, “Security attacks in iot: A survey,” International Conference on I-SMAC (I-SMAC), pp. 32–37, 2017. https://doi.org/10.1109/I-SMAC.2017.8058363

[2] Statista Inc., "Number of Internet of Things (IoT) connected devices worldwide from 2019 to 2030, by communications technology," Transforma Insights, December 2020 [https://www.statista.com/statistics/1194688/iot-connected-devicescommunications-technology]. [Online]. [Accessed 14-05- 2025].

[3] R. Uddin, & I. Koo, “Real-Time Remote Patient Monitoring: A Review of Biosensors Integrated with Multi-Hop IoT Systems via Cloud Connectivity”, Applied Sciences, vol. 14, 1876, 2024. https://doi.org/10.3390/app14051876

[4] M. Papaioannou, M. Karageorgou, G. Mantas,V. Sucasas, I. Essop, J. Rodriguez, & D. Lymberopoulos, “A Survey on Security Threats and Countermeasures in Internet of Medical Things (IoMT)”, Transactions on Emerging Telecommunications Technologies, vol. 33, e4049, 2022, https://doi.org/10.1002/ett.4049

[5] F. Hussain, S. Abbas, G.,Shah, A., Pires, I. Fayyaz, F. Shahzad, F., M. Garcia, & E. Zdravevski, “A framework for malicious traffic detection in IoT healthcare environment”, Sensors, vol. 21, 3025, 2021. https://doi.org/10.3390/s21093025

[6] S. Bhosale, M. Nenova, & G. Iliev, G., “A study of cyber attacks: In the healthcare sector”, In 2021 Sixth Junior Conference on Lighting (Lighting), pp. 1–6. https://ieeexplore.ieee.org/abstract/document/9598947

[7] S. Madanian, T. Chinbat, M. Subasinghage, D. Airehrour, F. Hassandoust, & S. Yongchareon, “Health IoT Threats: Survey of Risks and Vulnerabilities” Future Internet, vol. 16, 389, 2024a. https://doi.org/10.3390/fi16110389

[8] K. Sutradhar, R. Venkatesh, & P. Venkatesh, “Smart healthcare services employing quantum internet of things on metaverse”, In Healthcare Service in the Metaverse, pp. 170-189, 2024, CRC

[9] A. Kaur, & A. Jasuja, “Health monitoring based on IoT using Raspberry PI”, In 2017 International Conference on Computing, Communication and Automation (ICCCA), pp. 1335–1340, 2017. https://doi.org/10.1109/CCAA.2017.8230004

[10] M. Pham, Y. Mengistu, H. Do, & W. Sheng, “Delivering home healthcare through a cloud-based smart home environment (CoSHE)” Future Generation Computer Systems, vol. 81, pp. 129–140, 2018. https://doi.org/10.1016/j.future.2017.10.040

[11] D. Choi, H. Choi, & D. Shon, “Future changes to smart home based on AAL healthcare service”, Journal of Asian Architecture and Building Engineering, vol. 18, pp. 190–199, 2019 https://doi.org/10.1080/13467581.2019.1617718

[12] M. Chen, D. Cui, H. Haick, & N. Tang, “Artificial Intelligence-Based Medical Sensors for Healthcare System”, Advanced Sensor Research, vol. 3, 2300009, 2024, https://doi.org/10.1002/adsr.202300009

[13] K. Ahmed, M. Ismail, Z. Shehata, I. Djirar, S. Talbot, N. Ahmadzadeh, S. Shekoohi, S. Cornett, E. Fox, & A. Kaye, “Telemedicine, E-health, and multi-agent systems for chronic pain management”, Clinics and Practice, vo. 13, pp. 470–482, 2023. https://doi.org/10.3390/clinpract13020042

[14] S. Madanian, T. Chinbat, M. Subasinghage, D. Airehrour, F. Hassandoust, & S. Yongchareon, “Health IoT Threats: Survey of Risks and Vulnerabilities”, Future Internet, vol. 16, 389, 2024b. https://doi.org/10.3390/fi16110389

[15] M. Khan, & M. Alkhathami, “Anomaly detection in IoT-based healthcare: Machine learning for enhanced security”, Scientific Reports, vol. 14, 5872, 2024. https://doi.org/10.1038/s41598-024-56126-x

[16] A. Said, A.Yahyaoui, & T. Abdellatif, “Efficient anomaly detection for smart hospital IoT systems”, Sensors, vol. 21, 1026, 2021. https://doi.org/10.3390/s21041026

[17] K. Thilagam, A. Beno, A. Lakshmi, C. Wilfred, S. George, M. Karthikeyan, V. Peroumal, C. Ramesh, & P. Karunakaran, “Secure IoT Healthcare Architecture with Deep Learning-Based Access Control System”, Journal of Nanomaterials, vol. 2022, 2638613,2022. https://doi.org/10.1155/2022/2638613

[18] N. Alharbe, & M. Almalki, “IoT-enabled healthcare transformation leveraging deep learning for advanced patient monitoring and diagnosis”, Multimedia Tools and Applications, 2024. https://doi.org/10.1007/s11042-024-19919-w

[19] J. Pimple, & A. Sharma, “Enhancing Cyber-Physical System Security in Healthcare Through Ensemble Learning, Blockchain and Multi-Attribute Feature Selection”, In P. Dubey, M. Madankar, P. Dubey, & B. T. Hung (Eds.), The Impact of Algorithmic Technologies on Healthcare, 1st ed., pp. 349–373), 2025, Wiley. https://doi.org/10.1002/9781394305490.ch16

[20] Y. Saheed, & M. Arowolo, “Efficient cyber attack detection on the internet of medical things-smart environment based on deep recurrent neural network and machine learning algorithms”, IEEE Access, vol. 9, pp. 161546–161554, 2021. https://doi.org/10.1109/ACCESS.2021.3128837

[21] L. Ramasamy, F. Khan, M. Shah, B. Prasad, B, C. Iwendi, & C. Biamba, “Secure smart wearable computing through artificial intelligence-enabled internet of things and cyber-physical systems for health monitoring”, Sensors, vol. 22, 1076, 2022. https://doi.org/10.3390/s22031076

[22] I. Kilincer, F. Ertam, A. Sengur, R. Tan, & U. Acharya, “Automated detection of cybersecurity attacks in healthcare systems with recursive feature elimination and multilayer perceptron optimization”, Biocybernetics and Biomedical Engineering, vol. 43, pp. 30–41, 2023. https://doi.org/10.1016/j.bbe.2022.11.005

[23] M. Alsulami, “An AI-Driven Model to EnhanceSustainability for the Detection of Cyber Threats in IoT Environments”, Sensors, vol. 24, 7179, 2024. https://doi.org/10.3390/s24227179

[24] D. Prabakar,S. Qamar, & R. Manikandan, “Artificial intelligence–based security attack detection for healthcare cyber-physical system: Lightweight deep stochastic learning”, Securing Next-Generation Connected Healthcare Systems, pp. 51–70, 2024, Elsevier. https://doi.org/10.1016/B978-0-443-13951-2.00009-X

[25] Z. Liu, X. Jia, & B. Li, “RETRACTED ARTICLE: E-healthcare application cyber security analysis using quantum machine learning in

malicious user detection", Optical and Quantum Electronics, vol. 56, 476, 2024. https://doi.org/10.1007/s11082-023-05854-x

[26] D. Kavitha, & S. Thejas, "AI enabled threat detection: Leveraging artificial intelligence for advanced security and cyber threat mitigation", IEEE Access, vol. 12, pp. 173127 – 173136, 2024. http://doi.org/10.1109/ACCESS.2024.3493957

[27] A. Algarni, & V. Thayananthan, "Digital Health: The Cybersecurity for AI-based healthcare communication", vol. 13, pp.5858-5870, IEEE Access, 2025. https://doi.org/10.1109/ACCESS.2025.3526666

[28] S. Ksibi, F. Jaidi, & A. Bouhoula, "MLRA-Sec: An adaptive and intelligent cyber-security-assessment model for internet of medical things (IoMT)", International Journal of Information Security, vol. 24, pp. 1–20, 2025. https://doi.org/10.1007/s10207-024-00923-y

[29] R. Czekster, T. Webber, L. Furstenau, & C. Marcon, "Dynamic risk assessment approach for analysing cyber security events in medical IoT networks", Internet of Things, vol. 29, 101437, 2025. https://doi.org/10.1016/j.iot.2024.101437

[30] M. Akhi, C. Eising, & L. Dhirani, "TCN-Based DDoS Detection and Mitigation in 5G Healthcare-IoT: A Frequency Monitoring and Dynamic Threshold Approach", vol.13, pp. 12709 – 12733, 2025, IEEE Access. https://doi.org/10.1109/ACCESS.2025.3531659

[31] M. Hernandez-Jaimes, A. Martinez-Cruz, K. Ramírez-Gutiérrez, & C. Feregrino-Uribe, "Artificial intelligence for IoMT security: A review of intrusion detection systems, attacks, datasets and Cloud–Fog–Edge architectures", Internet of Things, vol. 23, 100887, 2023. https://doi.org/10.1016/j.iot.2023.100887

[32] K. Vijayakumar, K. Pradeep,A. Balasundaram, & M. Prusty, "Enhanced cyber attack detection process for internet of health things (IoHT) devices using deep neural network", Processes, vol. 11, 1072., 2023. https://doi.org/10.3390/pr11041072

[33] J. Maruthupandi, S. Sivakumar, B. Dhevi, S. Prasanna, R. Priya, & S. Selvarajan, "An intelligent attention based deep convoluted learning (IADCL) model for smart healthcare security". Scientific Reports, vol. 15, 136, 2025. https://doi.org/10.1038/s41598-024-84691-8

[34] A. Nasayreh, H. Khalid, H. Alkhateeb, J. Al-Manaseer, A. Ismail, & H. Gharaibeh, "Automated detection of cyber attacks in healthcare systems: A novel scheme with advanced feature extraction and classification", Computers & Security, 150, 104288, 2025. https://doi.org/10.1016/j.cose.2024.104288

[35] A. Berguiga, A. Harchay, & A. Massaoudi, "HIDS-IoMT: A deep Learning-Based intelligent intrusion detection system for the internet of medical thing", IEEE Access, vol. 13, pp. 32863 – 32882, 2025. https://doi.org/10.1109/ACCESS.2025.3543127

[36] E. Alalwany, B. Alsharif, Y. Alotaibi, A. Alfahaid, I. Mahgoub, & M. Ilyas, "Stacking Ensemble Deep Learning for Real-Time Intrusion Detection in IoMT Environments", Sensors, vol. 25, 624, 2025. ; https://doi.org/10.3390/s25030624

[37] M. Babar, M.Tariq, Z. Ullah, F. Arif, Z. Khan, & B. Qureshi, "An Efficient and Hybrid Deep Learning-Driven Model to Enhance Security and Performance of Healthcare Internet of Things", IEEE Access, vol. 13, pp. 22931 – 22945, 2025. https://doi.org/10.1109/ACCESS.2025.3536638

[38] Khan. M, Almulhim. A, Alkati, S, "Towards the evaluation of cybersecurity threats and challenges in higher education institutions in Saudi Arabia", Edelweiss Applied Scienc and Technology, vol. 9, pp. 657-669, 2025. https://doi.org/10.55214/25768484.v9i2.4564

[39] S. Dadkhah, E. Neto, R. Ferreira, R. Molokwu, S. Sadeghi, & A. Ghorbani, "CICIoMT2024: A benchmark dataset for multi-protocol security assessment in IoMT", Internet of Things, 28, 101351, 2024. https://doi.org/10.1016/j.iot.2024.101351

Performance Analysis of Proposed Scalable Reversible Randomization Algorithm (SRRA) in Privacy Preserving Big Data Analytics

Mohana Chelvan P[1], Dr. Rajavarman V N[2], Dr. Dahlia Sam[3]
Research Scholar, Department of Information Technology, Faculty of Engineering and Technology, Dr. M.G.R. Educational and Research Institute, Chennai, India[1]
Senior Assistant Professor, School of Science and Computer Studies, CMR University, Bengaluru, India[1]
Professor, Department of Information Technology, Faculty of Engineering and Technology, Dr. M.G.R. Educational and Research Institute, Chennai, India[2, 3]

***Abstract*—The economy of today's world is a data-driven knowledge economy, as electronic devices are mostly used for our day-to-day activities, through which organizations collect data actively or passively. The dimensionality of the dataset is also increased, along with the volume of data, because of the advancements in digital devices and communication technology. The feature selection becomes a crucial preprocessing step in big data analytics as a dimensionality reduction technique to eliminate redundant and noisy features. Studying the fluctuations in feature selection results is a vigorous area of research, as it is positively related to data utility, as fluctuations in feature selection results confuse the data analysts' minds about their research outcomes. Privacy preservation is a major concern in big data analytics to protect sensitive individuals' data. Application of privacy preservation techniques to modify the dataset will affect the stability of feature selection, as it has recently been proven that it mostly depends on the dataset's physical characteristics. This study analyses the performance of the proposed Scalable Reversible Randomization Algorithm (SRRA) in terms of privacy preservation, change in characteristics of the dataset, information loss, stability of feature selection, and data utility in big data scenarios.**

Keywords*—Big data; data analytics; high dimensionality; feature selection; selection stability; privacy preservation; information loss

I. Introduction

Big data is a keyword for organizations, as everyday mountains of customer data (including social media, education, healthcare, finance, business, and more) accumulate in organizations that are mostly unstructured and semi-structured. Because of the technological advancements and outsourcing of cloud computing, the dimensionality, i.e., the number of features of the dataset, is also increasing along with volume, i.e., samples. Organizations get an edge over their competitors by analyzing the data for mining patterns and getting insights, and applying the knowledge for business strategic decision-making and improved customer relationship management. However, feature selection is a vigorous preprocessing step of dimensionality reduction as it eliminates the "curse of high dimensionality" by selecting the most relevant subset of features for better comprehension of the model, higher efficiency in terms of processing time, improved accuracy of data analytics results, elimination of possible redundant and noisy attributes which may be quasi identifier attributes leading to better privacy preservation than the full set of features.

Unstable feature selection results confuse the researcher, as feature selection is related to data utility. Previously, data scientists believed that fluctuations in feature selection results depend on feature selection algorithms. However, lately, it has been proven that fluctuations in feature selection results depend mostly on changes in the physical characteristics of the dataset but are not completely independent of algorithms [1-6]. Feature selection instability is caused by minor changes in the dataset, which may select different subsets of features and may affect the accuracy of data analytics results, leading to doubts in the minds of researchers about their conclusions of research findings. Sensitive data like monetary data or disease, or marital status of individuals should be prevented from disclosure to third parties in data analytics. The application of privacy-preservation techniques affects the physical characteristics of the dataset, which will be reflected in fluctuations in feature selection results, which will be related to the accuracy of big data analytics results. Hence, there is a need to develop privacy preserving big data analytics algorithms with minimal change in the physical characteristics of the dataset for better feature selection stability results, lesser information loss, and higher data utility without compromising on privacy preservation.

All the privacy conserving methods like slicing, t-closeness, l-diversity, and k-anonymity have been practiced for a long time, but in the big data era, their applications have practical complications resulting in processing overheads because of the enormous volume, several varieties, and continuous velocity of arriving data. Traditional security mechanisms are inadequate in handling data analytics in big data and cloud computing scenarios. The tools for big data, like Hadoop, MapReduce, Pig, and Spark, work on distributed parallel processing in the Hadoop Distributed File System, leading to complications in employing privacy-conserving methods in terms of scalability and implementation overheads. It is practically very complex to implement and time-consuming in terms of processing efficiency, and it is very difficult to implement these privacy conserving methods in terms of scalability, as the data and processing algorithm are distributed in thousands of data nodes,

and also results in a diminution of the accuracy of outcomes of big data analytics, i.e., data utility.

Information security and privacy protection are very challenging in the big data and cloud computing era. To address the issues, the proposed algorithm, SRRA, is based on the randomization technique, which can be applied to the preprocessing stage itself; hence, it is highly scalable and efficient in terms of processing time for big data privacy preservation. At the preprocessing stage, the data are collected from different sources and go through data extraction, data cleansing, transformation procedures, and anonymization of sensitive and quasi-identifiers by SRRA to annihilate personal identifying information and safeguard private sensitive data, followed by feature selection as a dimensionality reduction technique.

In this research study, Section II provides the related work, and Section III describes the proposed privacy preserving big data analytics algorithm, the Scalable Reversible Randomization algorithm (SRRA). Section IV presents the methodology, results and discussion. Finally, Section V provides the conclusion.

II. Related Work

Big data analytics is vital for organizations in today's data-driven economy. However, sensitive data of individuals' records shouldn't be re-identified by the data analyst, as it can be misused. Privacy is a data owner-oriented concept and a privacy preservation method in which the dataset is under the control of a custodian called PPDM (Privacy Preserving Data Mining), and the data owner is responsible for the privacy of the individuals' data. The second one is the anonymization method, and also in this method, the dataset is not under the control of a custodian called PPDP (Privacy Preserving Data Publishing).

In PPDM, like differential privacy techniques is interactive, the dataset is under the control of the data owner, and the big data analyst analyses the dataset. However, these techniques are not appropriate for the big data era and are practically very difficult to implement because of the three v's, i.e., velocity, variety, and volume of big data, and also the data is collected from different sources. The PPDP methods like generalization, bucketization, perturbation, and anonymization techniques like k-anonymity (quasi identifiers are generalized to at least k number of alike attribute values for different records in each equivalence class), l-diversity (improved model of the k-anonymity technique, the sensitive attributes are at least l distinct domain feature values for each equivalence class that includes both generalized k number quasi identifier values and l distinct sensitive feature values to avoid both background knowledge and similarity attack), t-closeness (to eliminate skewness attack on l-diversity model), slicing, and randomization methods are non-interactive, the dataset is anonymized and published for any research works as it is not under the control of a custodian. However, these techniques are practically very difficult to implement in the distributed processing environment of big data and the cloud computing era of Hadoop, Spark, or Pig, as they are not directly applicable and also not very scalable.

The differential privacy method [7] is an interactive query made to a database using a randomized response mechanism by adding calculated noise to the outcome of the query. This method shares data to permit inferences about groups of persons while preventing someone from learning information about an individual through a rigorous mathematical foundation. It gives strong security for sensitive data even if the adversary has robust background knowledge, but less data utility because of added noise. Because of the three v's: variety, velocity, and volume of big data, the differential privacy technique is not appropriate for cloud computing and big data environments. In the Hadoop Distributed File System environment of big data, along with the distributed processing environment of the MapReduce algorithm, it is very complex and difficult to implement differential privacy in big data.

The important PPDP methods are t-closeness, l-diversity, k-anonymity, and slicing. The k-anonymity method anonymizes the data to eliminate re-identification, as every quasi-identifier is anonymized with a k number of feature values by the generalization technique [8, 9]. In the k-anonymity algorithm, the value of k of 23 means ensuring 23 blurry tuples when an effort is made to recognize a specific individual's private sensitive data. It is difficult to prevent re-identification of individuals' records by quasi-identifiers, and so disclosure of sensitive data cannot be prevented in k-anonymity. In the k-anonymity technique, record linkage attacks like homogeneity attacks and background knowledge attacks compromise the de-identification of an individual's record if there is insufficient diversity of sensitive attributes.

The l-diversity method was introduced to eliminate the attacks that compromise the k-anonymity model. There must be l well-represented sensitive feature values like disease, marital status, and monetary values in each equivalence class, in the l-diversity method [10]. Because of the different varieties and huge volume of data in big data, l-diversity is not possible at all times, and implementing l-diversity in big data is very complex and difficult. The feature disclosure cannot be safeguarded when the global distribution of data in the dataset is tilted into a few equivalence classes. Nevertheless, de-identification is compromised by the skewness attack in the l-diversity model, as in the equivalence class, one value emerges as frequently leading to the possibility for the opponent to deduce the value of another entity having a similar value with the information of real time global dispersal of data.

The t-closeness model is introduced to avoid the skewness attack in the l-diversity model. If the distance between the distributions of sensitive feature values in the class is no more than a threshold of the real skewness of the distributed data, the equivalence class is measured to have 't-closeness' in the t-closeness model [11]. These techniques have limitations and practical difficulties in big data and cloud computing scenarios. Applying these techniques is very difficult and complex in big data and clouds because of the huge volume and multiple varieties of data, and also inefficient due to long processing time, resulting in being very difficult to implement in big data frameworks like Hadoop, MapReduce, Pig, and Spark.

Slicing is the model of splitting the dataset column-wise or row-wise, i.e., splitting feature values and/or tuples where

extremely correlated features are a group and dissimilar features are sliced into different groups [12]. Slicing models are more appropriate for big data datasets with high dimensionality. Nevertheless, it is very complex to implement and also has practical limitations.

The randomization method adds calculated noise to the particular feature values [13]. Randomization methods have disadvantages in big data due to processing time efficiency and reduction in data utility, i.e., the accuracy of outcomes of big data analytics. However, the randomization method can be applied in the preprocessing step itself, and then it eliminates anonymization overhead.

Cryptographic methods such as secure multiparty computation (MPC) and homomorphic encryption are used to encrypt the data before the transfer of it for big data analytics [14]. In this method of secure multiparty computing, multiple parties can analyze the dataset together without revealing their data. It is very complex to implement cryptographic methods in big data. If multiple parties are involved in integrated big data analytics, managing the complexity is more challenging in big data. In this technique, data utility is considerably reduced by the application of encryption. Also, it is not easy in the big data and cloud computing scenario to encrypt all data.

The recent developments in privacy-conserving methods in the big data era are MapReduce-based anonymization (MRA) [15]. In this method, the dataset is considered as a single equivalence class, and in each iteration, tuples are selected for subclasses for k-anonymization. A global file is needed among the data nodes to share the updated equivalence class information with all nodes is a major drawback of the method. In the updated version of MRA, instead of a single global file, a chunk of files is used to update the equivalence class information. In this method, the drawbacks are multiple files to share equivalence class information and multiple iterations. The Scalable k-anonymization (SKA) model for MapReduce in [16]. This method has better processing time in comparison with similar existing scalable algorithms. In this method initial file is converted into a sorted equivalence class. The SKA algorithm results in considerable information loss. In the improved SKA, the columns are rearranged so that the distance between consecutive records is less, resulting in minimum information loss.

The privacy preserving big data analytics algorithm in [17] uses cryptographic techniques like homomorphic encryption schemes, attribute-based encryption schemes, and order-preserving encryption schemes in untrusted cloud environments. It is very difficult to implement this method in big data and cloud computing environments, as applying cryptographic techniques is a huge implementation overhead. This method has improved the preservation of privacy compared to other methods. The application of encryption algorithms for privacy preservation drastically reduces the utility of data.

Privacy preserving big data analytics algorithm in [18] uses two phases Named Entity Recognition (NER) approach called Conditional Random Field (CRF) to annotate unstructured data into the structured form, along with ImpSKA, an extension of the Scalable k-Anonymity (SKA) model using the Apache Pig framework. This method can be used for semi-structured and unstructured data. Practically, it is complex to implement in a big data distributed processing environment, leading to huge implementation overhead.

The improved scalable l-diversity approach in [19] is the scalable anonymization approach for privacy preserving big data publishing, and Apache Pig is used for the implementation. In this method, the input data are sorted for the creation initial equivalence class. In improved scalable l-diversity (ImSLD), both quasi-identifier attributes and sensitive attributes are used to generate an equivalence class from the sorted columns, and then the table is first k-anonymized and then l-diversified. In ImSLD, along with k-anonymization, l-diversity is also used to improve the preservation of privacy. There will be information loss and decreased data utility in comparison to the previous anonymization methods. It is a complex and time-consuming model in a big data scenario, as the required columns are arranged in ascending order of several unique values. This model also has a marginally higher processing time in comparison to similar models.

Along with generalization, anonymization (l-diversity and k-anonymity), slicing, and a method analogous to a one-way hash function called privacy view generation are used in the proposed system in [20] to safeguard the sensitive data and to avoid privacy incursions. In feature partitioning using the slicing method, extremely interrelated features are sliced as a subset of records. An equivalence class is formed for each slice of records and is anonymized by k-anonymity. Quasi identifiers are susceptible to background knowledge attacks, and hence sensitive feature values are l-diversified. However, this model is very complex to implement as it is very time-consuming in big data scenarios, leading to increased processing time and data utility is also drastically reduced.

The privacy-conserving big data analytics methods are not effective in terms of scalability, and applying these methods in big data distributed frameworks like Hadoop, MapReduce, Pig, or Spark is very intricate and time-consuming, and data utility is considerably reduced. The proposed Scalable Reversible Randomization Algorithm (SRRA) in this research study is suitable for the distributed parallel processing of big data frameworks, as it can be applied in the preprocessing stage itself, and so it is extremely scalable. It fills the gap, as the existing privacy preserving techniques are unsuitable in a big data scenario. The SRRA is practically not very difficult to implement in big data compared to the similar methods and with better fluctuations in feature selection results, lesser information loss, and higher data utility without compromising privacy preservation.

III. Proposed Scalable Reversible Randomization Algorithm (SRRA)

There will be four types of features in the dataset: non-sensitive features, indirect or quasi-identifiers, sensitive features, and direct identifiers.

Non-sensitive features are the features other than the above three types that are not sought by the intruder.

Indirect identifiers or quasi-identifiers are features like date of birth, occupation, religion, location, and zip code that can be

indirectly used for the re-identification of a record of an individual using linkage attacks.

Sensitive features are features like disease or monetary values targeted by the intruder. Privacy preservation techniques should be applied to quasi-identifiers and sensitive features to safeguard the privacy of individuals' records.

Direct Identifier attributes uniquely identify the record, like employee ID or social security number. Identifier attributes are generally anonymized before being sent for big data analytics.

The proposed reversible privacy conserving algorithm, i.e., Scalable Reversible Randomization Algorithm (SRRA), is shown in Fig. 1. SRRA is explained in [21]. Big data analytics is challenging for computational resources in terms of storing and processing, and also for machine learning algorithms in terms of efficiency and performance. Also, it is difficult to identify quasi-identifier and sensitive data in big data. The efficiency, i.e., performance of algorithms, and data utility, i.e., accuracy of results of the analysis, will be closer or the same because, with sampled versions, for the reduced samples selected by attribute selection optimization algorithms such as the ant-colony, Pareto-front, particle swarm optimization, and similar other algorithms. However, we should not be able to outperform the model of the 100% data. The privacy-conserving alteration by the proposed algorithm SRRA is scalable and efficient, along with better stability of feature selection and accuracy results.

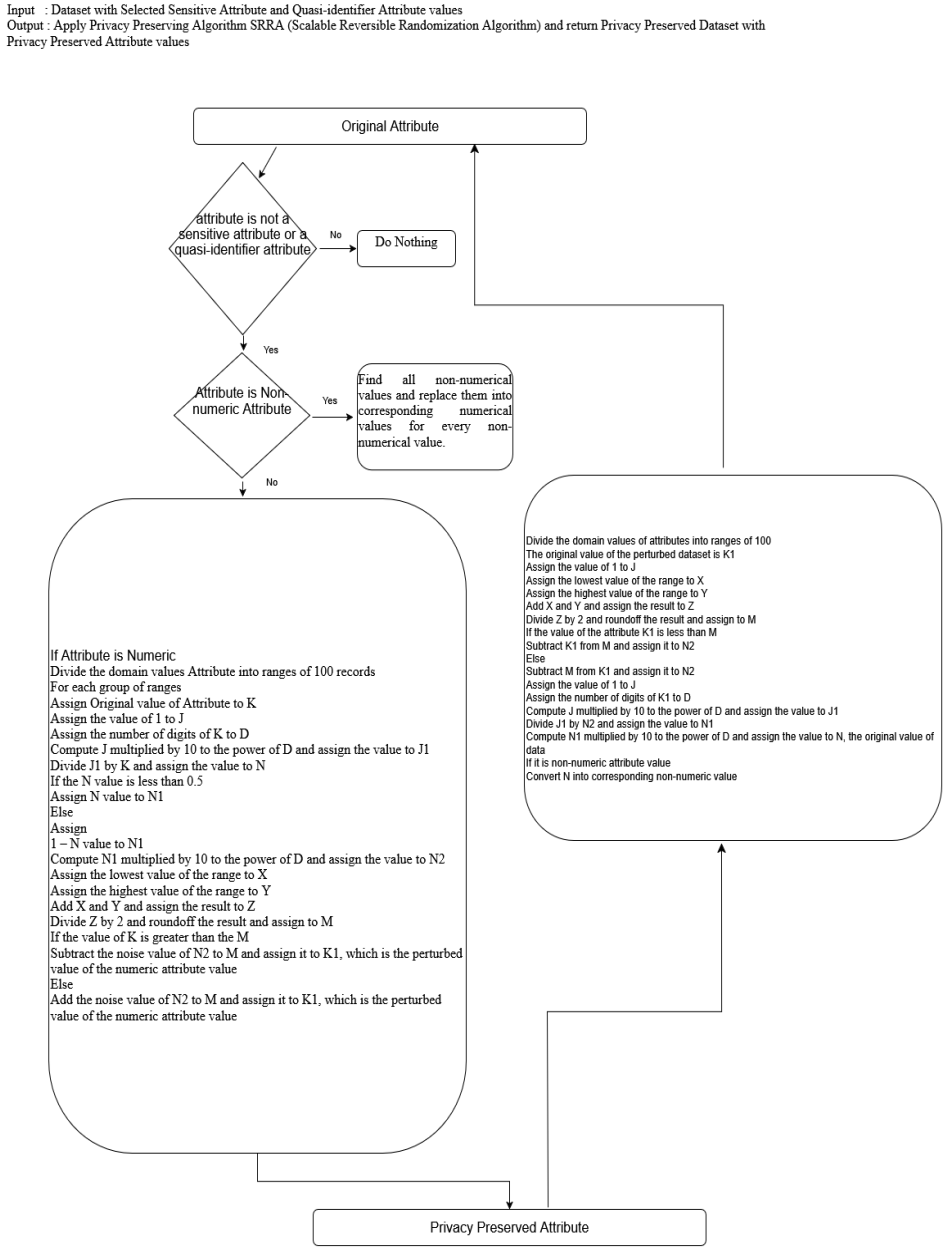


Fig. 1. Pseudocode of the proposed reversible privacy preserving algorithm of SRRA.

The selected sensitive and quasi-identifier attributes are perturbed by the proposed SRRA for privacy preservation. The application of SRRA in big data, which contains numeric and non-numeric data converted into definite sizes or formats, is suitable for the application of feature selection methods. The feature selection algorithm applied on the privacy conserved dataset, which is an important preprocessing step, will select an almost similar set of attributes, as the privacy conserving perturbation does not much affect the statistical characteristics of the experimental dataset.

The privacy-preserved dataset is used for big data analytics to get insights and hidden patterns from it. The application of SRRA results in the change in statistical characteristics of attribute values of the experimental dataset is negligible with less information loss, leading to better stability of feature selection and improved big data analytics results in terms of accuracy and error rate, along with robust privacy preservation of individuals' records. The SRRA processing time is reduced to around 15% in comparison to other similar methods in big data scenarios. The reversible privacy conserving algorithm of SRRA can be used to get the original dataset from the privacy-preserved dataset.

IV. Experiments

A. Methodology

The methodology of research is diagrammatically shown in the block diagram, in Fig. 2.

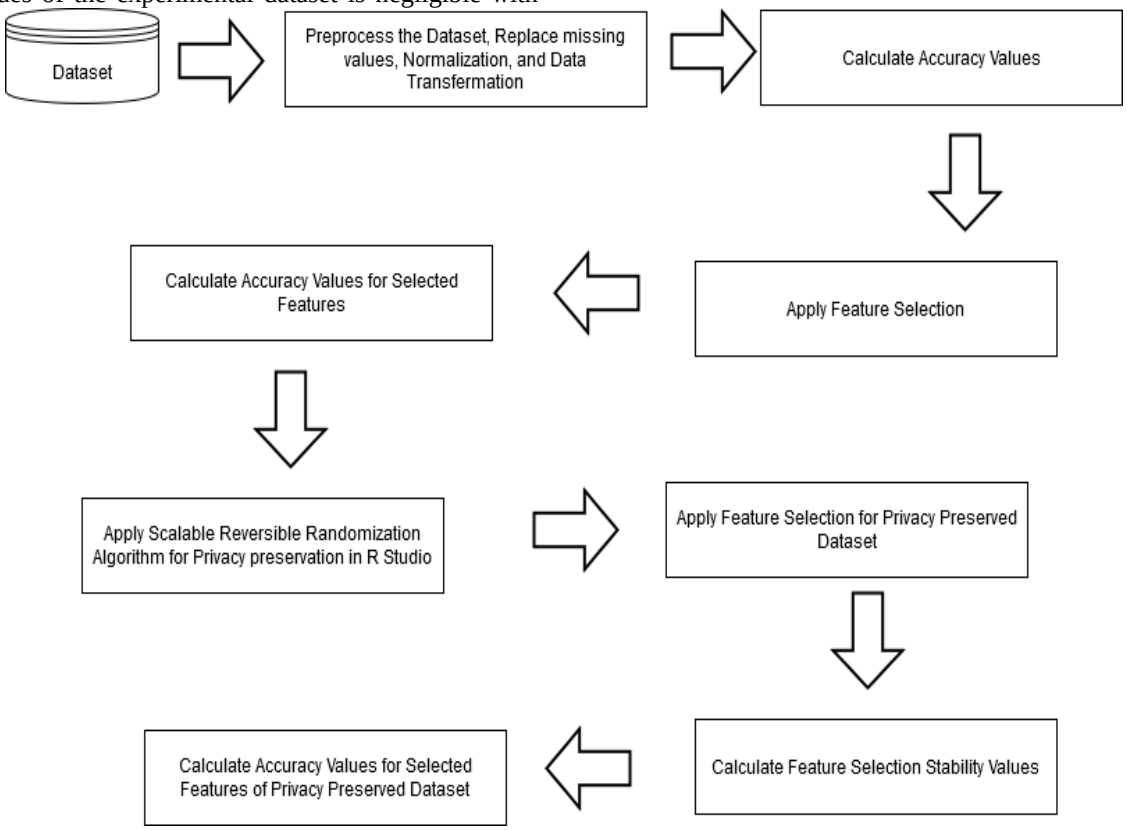


Fig. 2. Methodology.

B. Experimental Setup

The experiments are carried out in R-Studio. The application of privacy preserving modification by SRRA is implemented as a preprocessing step on the experimental datasets. The privacy-preserved datasets are harnessed in the Hadoop Distributed File System. The Hadoop distributed processing is implemented with three data nodes and one name node. The dataset used in the experiments is the Census Income dataset of the UCI repository.

C. Normalized Cardinality Penalty NCP

The metric to measure the information loss due to the manipulations of privacy-preservation is Normalized cardinality penalty (NCP) [22]. Information loss is an important parameter for the performance of the privacy preserving big data analytics algorithm. NCP is the metric of information loss in percentage, and hence it is suitable for big data scenarios. NCP for an equivalence class G for numeric attribute A is defined in Eq. (1):

$$NCP_A(G) = \frac{MAX_A^G - MIN_A^G}{MAX_A - MIN_A} \quad (1)$$

where, numerator MAX and MIN represent the range of attributes of class G, and denominator MAX and MIN represent the entire table.

D. Correlation-Based Feature Selection CFS

By seeing the degree of redundancy among them, along with the distinctive prophetic capability of each feature, the value of the attribute subsets is evaluated by CFS. The subsets of features that are prominently correlated within the class, with the inter-

correlation among classes, will be scanter preferences are taken by CFS [23]. Authors have GA, as a search method, with CFS as a fitness function. CFS selects the best feature subset and can be supplied with alternative search mechanisms. In Eq. (2), CFS is specified.

$$r_{zc} = \frac{k\, r_{zi}}{\sqrt{k + (k-1)\, r_{ii}}} \quad (2)$$

where, k refers to the number of subset features, r_{ii} refers to the mean inter-correlation between subset features, r_{zi} refers to the mean correlation between subset features and the class variable, and r_{zc} provides the correlation between other subsets of features and the class variable [23].

E. Dice's Coefficient

In [24], the Dice coefficient is the similarity measure used to measure the stability of feature selection, i.e., to calculate the overlap between two subsets. It included the Jaccard index. Dice takes a value between zero and one, where zero suggests no overlap and one suggests that the two subsets are identical. Dice, Jaccard, and Tanimoto are measures of similarity among them to calculate fluctuation in feature selection results. The Dice's coefficient is represented in the given Eq. (3):

$$\text{Dice}(F'1, F'2) = \frac{2\,|F'1 \cap F'2|}{|F'1| + |F'2|} \quad (3)$$

F. Results

In the experiments, four algorithms from Table I are used. Dice's coefficient is a metric used to measure fluctuations in feature selection results. The feature selection algorithm used in the experiment is CFS. Table I shows the relationship between changes in the physical characteristics of the dataset by privacy preserving perturbation effects on information loss, the fluctuations in feature selection results, and the accuracy of big data analytics outcomes.

In [20], privacy preservation is implemented in the HDFS distributed file system environment of real-time huge datasets. A secure map-reduce layer is introduced in MapReduce distributed processing for privacy preserving implementation. The methodologies used are generalization, slicing, anonymization (k-anonymity followed by l-diversity), and a one-way hash function, as privacy view generation in a parallel processing environment of distributed processing. In this method, the l value is kept constant for changing values of k for different anonymization views and processing time combinations for optimum configuration values. However, by using NCP, it is proven that information loss in 48% results in decreased data utility.

The algorithm is a combination of techniques like slicing, generalization, k-anonymity, l-diversity, one way hash function resulting in better privacy-preservation, but implementing this algorithm in the Hadoop distributed file system is very complex and time consuming, resulting in increased processing overhead in big data scenario. Stability of feature selection is 0.72 as it is drastically reduced along with data utility value for overall accuracy as 63.27% and accuracy of selected features as 68.59%.

TABLE I PERFORMANCE COMPARISON OF DIFFERENT TECHNIQUES

Privacy Preserving Technique Ref. No.	Information loss by NCP	Feature Selection Stability	Overall Accuracy		Accuracy of Selected Features	
			Before Privacy Preservation	*After Privacy Preservation*	*Before Privacy Preservation*	*After Privacy Preservation*
[20]	48%	0.72	74.51%	63.27%	79.68%	68.59%
[19]	35%	0.86	74.51%	69.68%	79.68%	75.51%
[18]	41%	0.77	74.51%	65.89%	79.68%	73.23%
[17]	59%	0.67	74.51%	61.87%	79.68%	65.86%
SRRA	26%	0.91	74.51%	72.28%	79.68%	77.41%

The privacy preserving algorithm by the authors in [19] is an improved scalable l-diversity approach, which has a higher stability of feature selection value of 0.86 and accuracy of overall feature as 69.68% and accuracy of selected features as 75.51% but it has lower privacy preservation compared to other methods. The information loss by NCP is 35%. The algorithm by [18] can be applied for semi-structured data using tagging techniques followed by scalable k-anonymity. The model is optimum in stability value of feature selection as 0.77, information loss by NCP as 41% and data utility of overall features as 65.89% and for selected features as 73.23%.

The algorithm by [17] is a cryptographic method, and it has the lowest stability of feature selection value of 0.67 and an overall accuracy value of 61.87% along with the accuracy of selected features as 65.86%, as it uses encryption methods, but it has better privacy preservation. In this method, the information loss by NCP is 59%, which is the highest among the listed methods.

The SRRA algorithm in [21] uses a randomization technique of calculating noise addition. The algorithm is applicable for both numeric and non-numeric data and hence suitable for big data. If the data is categorical or non-numeric, the attribute values are converted into equivalent numeric values before applying the algorithm. The proposed algorithm SRRA has the highest stability of feature selection value of 0.91 compared to other algorithms, an overall accuracy of 72.28%, and an

accuracy of selected features of 77.41%, which is higher than other algorithms. The information loss by NCP is also a minimum value of just 26%. This is because the change in statistical properties by SRRA is minimal compared to other algorithms, as the change in physical characteristics of the trial dataset is positively related to fluctuations in feature selection results and accuracy.

G. Discussion

The method in [20] is superior in terms of safeguarding private sensitive data. This method is very complex to implement in terms of implementation and computing cost, i.e., execution time. Implementation of the algorithm in [19], in the Hadoop distributed file system, using the MapReduce algorithm, is also easy compared to other models. This model in [18] is difficult to implement compared to other models used in the experiments.

Encryption methods in [17] are difficult to implement in big data scenarios, and data utility is also drastically reduced. So, there will be a relationship among privacy preserving perturbation, information loss, fluctuations in feature selection results, and accuracy.

As the characteristics of the dataset are related to the fluctuations in feature selection results and accuracy, the statistical characteristics of the dataset have minimal modification because of the privacy-conserving alteration by SRRA in [21]. Hence, SRRA give better results in terms of stability of feature selection, accuracy values of privacy-preserved dataset in terms of overall accuracy, and accuracy of selected features.

Information loss by NCP is also minimum, compared to other similar methods, as the change in characteristics of the dataset is minimal. Another advantage of SRRA is its suitability in big data scenarios, as the privacy preserving algorithm is a simple calculated noise addition technique and can be implemented in the preprocessing stage itself with less overhead, i.e., processing time, compared to similar privacy preserving algorithms without compromising on privacy-preservation.

V. Conclusion

The SRRA has higher stability of feature selection and accuracy values because the change in characteristics of the physical properties of the experimental dataset is minimal. However, the algorithm has better privacy preservation as there is a trade-off among variation in physical characteristics of the dataset, information loss, efficiency, fluctuations in feature selection results, as it is related with data utility, i.e. accurate data analytics results.

In the proposed privacy preserving big data analytics algorithm SRRA, both quasi-identifiers and sensitive features are anonymized by the randomization technique of calculating noise addition efficiently with minimal practical overheads in the preprocessing stage itself, and hence, because of the practical advantages, the implementation overhead is minimal compared to similar other algorithms in big data and cloud environments.

General privacy preserving algorithms are not working in a big data scenario. Hence, developing privacy-preservation algorithms in a big data scenario is very much challenging because of the implementation overhead and hence there will be a tradeoff among lesser information loss, better privacy preservation, improved stability of feature selection as it is directly linked with data utility, as selected features are used for data analytics for better efficiency, i.e. processing time and improved accuracy of data analytics results.

References

[1] Salem Alelyani, Huan Liu., The Effect of the Characteristics of the Dataset on the Selection Stability, IEEE DOI 10.1109/International Conference on Tools with Artificial Intelligence, 2011.167, 1082-3409/11, http:// ieeexplore.ieee.org/document/ 6103458, 2011.

[2] Salem Alelyani, Zheng Zhao, Huan Liu., A Dilemma in Assessing Stability of Feature Selection Algorithms, IEEE DOI 10.1109/ International Conference on High Performance Computing and Communications, 2011.99, 978-0-7695-4538-7/11, http:// ieeexplore.ieee.org/document/6063062, 2011.

[3] Salem Alelyani, On feature selection stability: a data perspective, Doctoral Dissertation, Arizona State University, AZ, USA, ISBN: 978-1-303-02654-6, ACM Digital Library, 2013.

[4] Barbara Pes, Feature Selection for High-Dimensional Data: The Issue of Stability, Proceedings of the 26th IEEE International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE 2017), June 21–23, 2017.

[5] Jundong Li, Huan Liu, Challenges of Feature Selection for Big Data Analytics, Special Issue on Big Data, IEEE Intelligent Systems, eprint arXiv:1611.01875, 2017.

[6] Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P. Trevino, Jiliang Tang, and Huan Liu, Feature Selection: A Data Perspective. ACM Comput. Surv, 50, 6, Article 94, 45 pages. DOI: https://doi.org/10.1145/3136625, 2018.

[7] C. Dwork, Differential privacy, Proceedings of the 33rd international conference on Automata, Languages, and Programming - Volume Part II, Pages 1–12 https://doi.org/10.1007/11787006_1pages 1-12, July 2006.

[8] Bayardo RJ, Agrawal A. Data privacy through optimal k-anonymization. In: Proceedings 21st international conference on data engineering, 2005 (ICDE 2005). Piscataway: IEEE; 2005.

[9] LeFevre K, DeWitt DJ, Ramakrishnan R. Incognito: efficient full-domain k-anonymity. In: Proceedings of the 2005 ACM SIGMOD international conference on management of data. New York: ACM; 2005.

[10] Machanavajjhala A et al. L-diversity: privacy beyond k-anonymity. In: Proceedings of the 22nd international conference on data engineering (ICDE'06), 2006. Piscataway: IEEE; 2006.

[11] Anil Prakash, Ravindar Mogili, Privacy Preservation Measure using t-closeness with combined l-diversity and k-anonymity, International Journal of Advanced Research in Computer Science and Electronics Engineering (IJARCSEE) Volume 1, Issue 8, pp:28-33, 2012.

[12] Tiancheng Li, Jian Zhang, Ian Molloy.: Slicing: A New Approach for Privacy Preserving Data Publishing. IEEE Transaction on KDD, 2012.

[13] Aggarwal CC, Philip SY. A general survey of privacy-preserving data mining models and algorithms. Privacy-preserving data mining. Springer: US; p. 11–52., 2008.

[14] Jiang R, Lu R, Choo KK. Achieving high performance and privacy-preserving query over encrypted multidimensional big metering data. Future Gen Comput Syst. 78:392–401., 2018.

[15] Mehta, B.B., Rao, U.P., Privacy preserving big data publishing: a scalable k anonymization approach using MapReduce. IET Software 11, 271–276. https://doi.org/10.1049/iet-sen.2016.0264., 2017.

[16] Mehta, B.B., Rao, U.P., 2018. Toward scalable anonymization for privacy preserving big data publishing. Recent Findings Intell. Comput. Tech. 708, 297–304.https://doi.org/10.1007/978-981-10-8636-6. Proceedings of the 5th ICACNI 2017, vol. 2., 2017.

[17] H. Shekhawat, S. Sharma and R. Koli, "Privacy-Preserving Techniques for Big Data Analysis in Cloud," Second International Conference on

Advanced Computational and Communication Paradigms (ICACCP), Gangtok, India, pp. 1-6, doi: 10.1109/ICACCP.2019.8882922, 2019.

[18] B. Mehta, U. P. Rao, R. Gupta and M. Conti, "Towards privacy preserving unstructured big data publishing", J. Intell. Fuzzy Syst., vol. 36, no. 4, pp. 3471-3482, 2019.

[19] Brijesh B. Mehta, Udai Pratap Rao, "Improved l-diversity: Scalable anonymization approach for Privacy Preserving Big Data Publishing", Journal of King Saud University – Computer and Information Sciences 34, 1423–1430, 2022.

[20] Ganesh Dagadu Puri and D. Haritha, "Implementation of Big Data Privacy Preservation Technique for Electronic Health Records in Multivendor Environment" International Journal of Advanced Computer Science and Applications (IJACSA), 14(2), (DOI): 10.14569/IJACSA.2023.0140214, 2023.

[21] Mohana Chelvan P, Rajavarman V N, "The Scalable Reversible Randomization Algorithm (SRRA) for better privacy preservation, improved feature selection stability, and higher accuracy in Big Data Analytics" Journal of Theoretical and Applied Information Technology, ISSN: 1992-8645, E-ISSN: 1817-3195, Volume 103, Issue 03, pp. 794 – 801, 2025.

[22] Ghinita, G., Karras, P., Kalnis, P., Mamoulis, N., Fast data anonymization with low information loss. In: Proceedings of the 33rd International Conference on Very Large Data Bases, VLDB '07, VLDB Endowment. VLDB Endowment, Vienna, Austria, pp. 758–769. https://doi.org/10.1109/ICDE.2007.369025., 2007.

[23] Mark A. Hall, Correlation-based feature selection of discrete and numeric class machine learning, Dept of Computer science, University of Waikato, 2000.

[24] Lei Yu, Chris Ding, and Ste1ven Loscalzo, Stable feature selection via dense feature groups. In KDD '08: Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 803–811, New York, NY, USA, 2008.

Shared API Call Insights for Optimized Malware Detection in Portable Executable Files

Mehdi Kmiti, Jallal Eddine Moussaoui, Khalid El Gholami, Yassine Maleh
LaSTI Laboratory-National School of Applied Sciences Khouribga, Sultan Moulay Slimane University, Beni Mellal, Morocco

***Abstract*—Malware analysis is essential for understanding malicious software and developing effective detection strategies. Traditional detection methods, such as signature-based and heuristic-based approaches, often fail against evolving threats. To address this challenge, this study proposes a static analysis–based malware detection system that employs thirteen classifiers, including Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Naive Bayes, Decision Tree, Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), Random Forest, Extra Trees, Gradient Boosting, AdaBoost, and LightGBM. The framework is built on a balanced dataset of 1,318 Windows Portable Executable (PE) files (674 malware, 644 benign), where the features are derived from shared API calls between benign and malicious files to ensure relevance and reduce redundancy. Experimental results show that the Extra Trees classifier achieved the highest accuracy of 98.14%, highlighting its effectiveness in detecting malware. Overall, this study provides a robust, data-driven approach that enhances static malware detection and contributes to strengthening cybersecurity against emerging threats.**

***Keywords*—Malware detection; static analysis; portable executable (PE) files; API calls; extra trees classifier**

I. Introduction

Malware, short for "malicious software", represents a major threat to digital systems, ranging from financial losses and privacy breaches to operational disruptions in critical infrastructure [1]–[4]. Modern malware is increasingly sophisticated, driven by profit, espionage, and cyber warfare, and includes forms such as viruses, worms, trojans, ransomware, and spyware [5], [6]. Recent statistics highlight its scale: more than 90% of digital threats target the Windows operating system, with AV-TEST reporting 1.17 billion threats by mid-2024, while global cybercrime costs are projected to reach $10.5 trillion annually by 2025 [7], [8]. These figures underscore the urgent need for effective malware detection strategies.

Traditional methods, such as signature-based and heuristic-based detection, struggle to keep pace with polymorphic and zero-day malware [9]–[11]. As a result, static analysis, which inspects executables without execution, has emerged as a promising alternative, particularly when combined with machine learning for classification.

Research Gap: Despite promising results, many prior static analysis methods rely on large and redundant feature sets, which increase computational overhead and reduce interpretability. Deep and ensemble learning models achieve high accuracy, but often lack scalability in real-world or resource-constrained environments. There is thus a need for lightweight and interpretable approaches that preserve strong detection performance while reducing redundancy.

To address this gap, we propose a malware detection framework based on shared API calls, which represent the intersection of calls found in both benign and malicious Portable Executable (PE) files. This refined feature space highlights the most relevant behaviors, reduces dimensionality, and supports efficient classification. Using a balanced dataset of 1,318 PE files (674 malware, 644 benign), we evaluate thirteen classifiers and demonstrate that the Extra Trees ensemble achieves the highest accuracy of 98.14%. The contributions of this work lie in constructing a refined dataset of shared API calls that enhances interpretability and efficiency, conducting a comparative evaluation of multiple classifiers, and showing that Extra Trees provides the best trade-off between accuracy and computational cost, making it suitable for practical deployment.

The remainder of this study is organized as follows: Section II reviews related work, covering metadata-based techniques, ensemble and hybrid approaches, and recent deep learning and graph-/transformer-based solutions, followed by the positioning of this study. Section III describes the proposed methodology, including dataset construction, API call extraction, shared-feature representation, preprocessing, the employed machine learning models, training and evaluation protocols, and the computing environment. Section IV presents the experimental results and provides a comprehensive discussion, including a comparative assessment against state-of-the-art (SOTA) approaches, limitations of the proposed method, practical applicability, and future research directions. Finally, Section V concludes the study by summarizing the key findings and outlining directions for future research.

II. Related Work

Research on static malware detection has explored a wide range of features and machine learning models. Existing approaches can be grouped into three main categories: metadata-based methods, ensemble and hybrid models, and deep learning methods, including recent transformer and graph-based solutions.

A. PE Header and Metadata Features

Balram et al. [12] evaluated six classifiers, including Support Vector Machine, Logistic Regression, Random Forest, and XGBoost, on string-based and PE header features. Their findings showed that string features, when combined with XGBoost and hybrid models, provided superior accuracy.

Similarly, Baldangombo et al. [13] analyzed PE headers, DLL names, and API calls, reporting that PE header features alone could achieve up to 99% accuracy with the J48 classifier while maintaining low computational overhead. These studies highlight the effectiveness of handcrafted metadata features but also underscore their limitations in scalability and adaptability to evolving threats.

B. Ensemble and Hybrid Models

Azeez et al. [14] proposed an ensemble framework that integrated fully connected dense Artificial Neural Networks (ANNs) and 1D-CNNs, with Extra Trees serving as a final-stage classifier. Their approach outperformed conventional classifiers in accuracy and robustness. Shijo and Salim [15] combined static and dynamic analysis, using Printable Strings Information (PSI) along with API call sequences extracted from the Cuckoo Sandbox, and achieved 98.7% accuracy with Support Vector Machine (SVM) and Random Forest. Yousuf et al. [16] further improved performance by integrating multiple feature sets, including DLL names, API calls, PE headers, and section attributes, and applying ensemble techniques such as Majority Voting, Stack Generalization, and AdaBoost, achieving 99.5% accuracy. These works demonstrate the benefits of combining diverse features and models, though often at the cost of increased complexity.

C. Deep Learning and Graph-/Transformer-Based Methods

Deep learning techniques have also been widely adopted in malware detection. Bensaoud and Kalita [17] proposed a CNN–LSTM model trained on 8-gram sequences, achieving an SOTA accuracy of 99.91%. More recent studies (2021 to 2025) have introduced advanced architectures based on transformers and graph neural networks (GNNs). Trizna et al. [18] applied a Transformer model with self-attention to dynamic API-call behaviors, improving detection accuracy while enhancing interpretability through attention visualization. Similarly, Seneviratne et al. [19] leveraged Vision Transformers in a self-supervised setup on image-based malware representations, achieving around 97% accuracy.

In the graph-based domain, GNNs are increasingly used to capture structural relationships in executables. Mohammadiana et al. [20] combined graph reduction techniques with GNNExplainer to enable efficient and interpretable malware detection on function and control-flow graphs. Shokouhinejad et al. [21] proposed a stacked GNN framework where multiple base learners are trained on PE control-flow graphs and integrated via an attention-based meta-learner, significantly improving both classification accuracy and explainability. Additionally, Tang [22] introduced a hybrid GNN-based model that fuses static and dynamic features, reporting a 4 to 7% performance gain over existing baselines on the EMBER and VirusShare datasets.

These works demonstrate the potential of deep learning, transformers, and GNNs for achieving high detection accuracy while also improving interpretability. However, their computational requirements are often high, making them less practical for deployment in resource-constrained environments.

D. Positioning of this Work

While prior studies demonstrate that traditional machine learning, ensembles, and deep learning can achieve high malware detection accuracy, they often rely on large or redundant feature sets or demand high computational resources. In contrast, the proposed framework focuses on a lightweight subset of shared API calls between benign and malicious executables. This design reduces redundancy, improves interpretability, and provides a practical balance between accuracy and efficiency, making it suitable for real-world deployment in static malware detection.

III. Methodology

This section details the data sources and selection criteria, the API-call extraction pipeline, the construction of the shared-call feature space, the preprocessing steps, the learning models and hyperparameters, and the training and evaluation protocol used in this study. The objective is to provide a complete and reproducible description of the static-analysis workflow. Table I presents the dataset overview.

A. Data Sources and Selection Criteria

The corpus comprises 1,318 Windows PE files: 674 malware samples obtained from VirusShare (2012 to 2024) and 644 benign executables collected from the C:\Windows\System32 directory of a Windows 10 installation. Duplicates at the file and feature-vector levels were removed. To limit class bias and facilitate clear performance estimation, the dataset was balanced via random undersampling of the majority class, yielding 714 unique samples (357 benign, 357 malware). The final feature matrix contains 559 columns: 558 binary API-call indicators and one label column. Rationale: a balanced setting supports robust classifier comparison without prevalence-driven bias, while focusing on static features enables scalable, execution-free analysis [23]-[27].

TABLE I Dataset Overview

_c_exit	SetLastError	VariantUnit	CopyFileW	…	Class
1	0	0	1	…	1
0	0	1	0	…	1
1	1	0	1	…	0
0	1	1	1	…	0
0	0	1	0	…	1

B. API-Call Extraction Pipeline

API calls were obtained through a two-stage static workflow:

1) Import table parsing using pefile to enumerate imported functions and their DLLs; normalization unified function aliases (e.g., CopyFileA/W) and handled forwarded exports.

2) Disassembly using the Capstone framework to scan code sections for call instructions and resolve additional API references not declared in the import table (e.g., late binding, indirect calls). Extracted names were canonicalized to DLL.Function form (e.g., KERNEL32.CopyFileW), lower-cased, and filtered to exclude non-WinAPI stubs and compiler intrinsics. The output of this stage is a per-file multiset of API

call identifiers [28]-[35]. Fig. 1 shows the steps of data collection process.

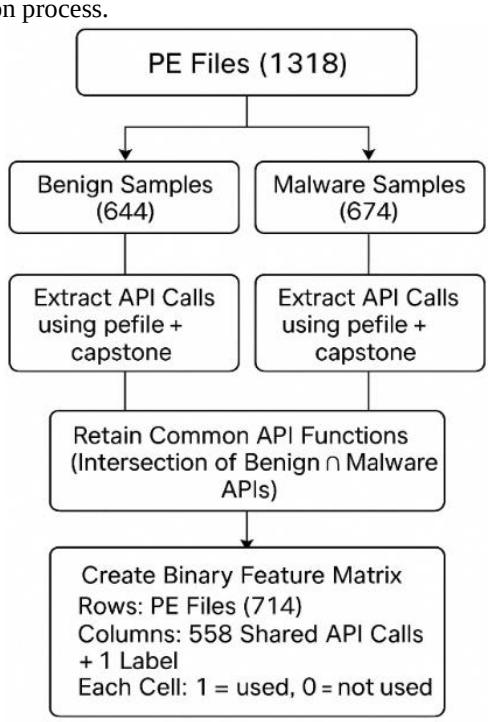


Fig. 1. Data collection process steps.

C. Shared-Call Feature Construction

Let B and M denote the sets of unique API calls observed in benign and malware subsets, respectively. The shared feature set is defined as:

$$S = B \cap M$$

Each file i is represented by a binary vector $x_i \in \{0,1\}^{|S|}$, where $x_{i,j} = 1$ if API $s_j \in S$ appears in file i, and 0 otherwise. Stacking all N files yields $X \in \{0,1\}^{N \times |S|}$. Limiting features to S reduces redundancy and emphasizes behaviorally relevant calls common to both classes.

Pseudo-code (shared-call construction)

1) Extract API sets A_f for each file f
2) Compute $B = \bigcup_{f \in \text{benign}} A_f$, $M = \bigcup_{f \in \text{malware}} A_f$
3) Set $S = B \cap M$
4) For each file f, form $x_f[j] = 1\{s_j \in A_f\}$ for all $s_j \in S$
5) Concatenate all x_f to obtain X.

a) Practical advantage: This representation filters out API calls unique to one class, thereby reducing dimensionality, mitigating dataset sparsity, and emphasizing behaviorally significant features. Compared with using the full API space, this method enhances computational efficiency, supports faster training, and highlights generalizable patterns for malware detection.

D. Preprocessing

Duplicate vectors were removed to ensure one instance per unique API profile. Missing indicators were treated as absence (0). The dataset was then shuffled and split into 70% training / 30% testing with stratification by class. No numerical scaling was applied to binary indicators. Where appropriate (e.g., SVM, LR), class weights were set to balanced [36]-[42]. Fig. 2 presents the data preprocessing steps.

E. Machine Learning Models and Parameters

To evaluate the effectiveness of shared-API-call-based static malware detection, a diverse set of classification algorithms was implemented, covering traditional learners, ensemble methods, and neural networks. Each classifier was trained on the binary feature vectors derived from Section III-C, with performance assessed via 5-fold stratified cross-validation and a 30% independent test set. Hyperparameters were selected based on widely adopted defaults or light grid search to balance performance and computational efficiency.

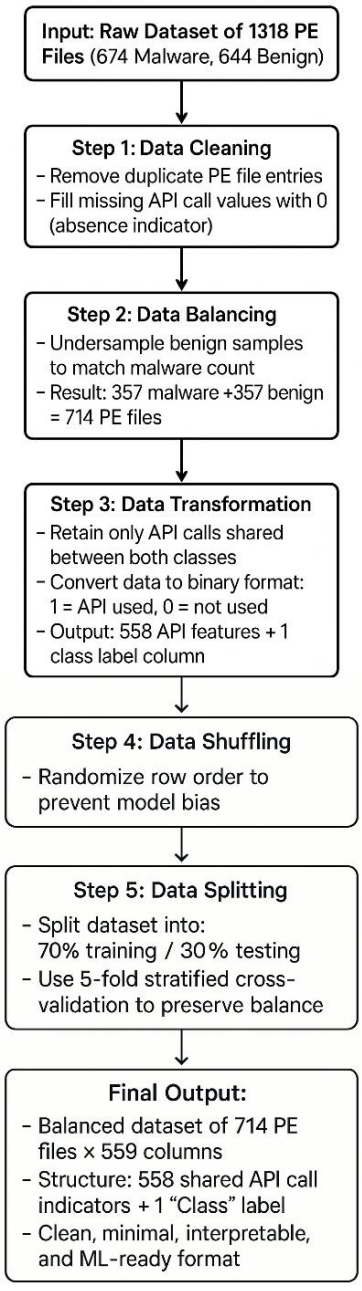


Fig. 2. Data preprocessing steps.

The classifiers are grouped into three categories: traditional baselines, ensemble learners, and deep learning models. A summary of libraries and configurations are presented in Table II to Table IV.

1) Traditional machine learning models: These models serve as interpretable baselines, providing insight into linear separability and low-complexity decision boundaries.

TABLE II TRADITIONAL MACHINE LEARNING MODELS

Model	Library	Hyperparameters / Notes	Practical Rationale
Logistic Regression	scikit-learn	solver='liblinear', penalty='l2', C=1.0, random_state=42	Robust linear baseline; efficient on sparse binary features
K-Nearest Neighbors (KNN)	scikit-learn	n_neighbors=5, weights='uniform', metric='minkowski'	Captures local neighborhoods; sensitive to feature sparsity
SVM	scikit-learn	C=1.0, kernel='rbf', gamma='scale'	Strong nonlinear decision boundary; benchmark for high-dimensional data
Naive Bayes	scikit-learn	Default parameters	Probabilistic, fast baseline for binary data
Desicion Tree	scikit-learn	criterion='gini', random_state=42	Provides interpretability; prone to overfitting
Linear Discriminant Analysis (LDA)	scikit-learn	Default parameters	Tests linear discriminants; assumes Gaussian feature distribution
Quadratic Discriminant Analysis (QDA)	scikit-learn	Default parameters	Quadratic boundary; serves as generative baseline

2) Ensemble learning models: Ensembles aggregate multiple weak learners to improve generalization. These models are particularly suited for binary, high-dimensional spaces, like API call vectors.

TABLE III ENSEMBLE LEARNING MODELS

Model	Library	Hyperparameters / Notes	Practical Rationale
Random Forest	scikit-learn	n_estimators=100, random_state=42	Reduces variance via bagging; robust baseline
Extra Trees	scikit-learn	n_estimators=100, random_state=42	Increased randomization; faster training
Gradient Boosting	scikit-learn	n_estimators=100, learning_rate=0.1, random_state=42	Strong baseline for structured data; balances bias/variance
AdaBoost	scikit-learn	n_estimators=50, learning_rate=1.0, random_state=42	Emphasizes difficult samples; reduces bias
LightGBM	LightGBM	Default parameters	Gradient boosting optimized for large sparse binary features

3) Deep learning models: Deep learning models are increasingly applied in malware detection due to their ability to capture complex, nonlinear relationships among features. In this study, a feedforward ANN was implemented as a representative deep learning baseline. The ANN directly ingests binary API-call feature vectors, enabling it to learn higher-order dependencies beyond the capacity of linear or tree-based classifiers.

TABLE IV DEEP LEARNING MODELS

Model	Library	Hyperparameters / Notes	Practical Rationale
ANN	Keras / TensorFlow	• Input: 559 nodes (one per API call) • Hidden layers: [256, 128], activation='relu' • Dropout: 0.5 • Output: 1 node, activation='sigmoid' • Optimizer: Adam • Loss: Binary Crossentropy • Epochs: 50 • Batch size: 32	Baseline neural network for binary feature vectors

F. Training and Evaluation Protocol

The dataset was partitioned into 70% training and 30% testing with stratified sampling to preserve class balance. Model selection and stability were assessed using 5-fold stratified cross-validation on the training portion; final results were reported on the independent test set. Unless otherwise stated, a fixed threshold of 0.5 on the predicted probability/score was used to compute threshold-dependent metrics (OA, TNR, FPR). AUC was computed from the ROC curve and is threshold-independent. All experiments used identical splits and fixed random seeds for reproducibility.

Metrics. Four complementary measures were reported:

a) Overall Accuracy (OA): proportion of correctly classified samples.

b) True Negative Rate (TNR): proportion of benign files correctly identified as benign.

c) False Positive Rate (FPR): proportion of benign files misclassified as malware.

d) Area Under the ROC Curve (AUC): probability that a randomly chosen malware sample receives a higher score than a randomly chosen benign sample; operationally computed from the ROC via the trapezoidal rule.

All equations are referenced:

$$OA = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \quad (1)$$

$$TNR = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \quad (2)$$

$$FPR = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} = 1 - TNR \quad (3)$$

$$AUC = \int_0^1 TPR(\tau) \times \mathrm{dFPR}(\tau) \ , With\ TPR = \frac{\mathrm{TP}(\tau)}{\mathrm{TP}(\tau) + \mathrm{FN}(\tau)} \quad (4)$$

G. Reproducibility and Computing Environment

The experiments were conducted on a workstation equipped with an Intel Core i5-12450H CPU (8 cores, up to 4.4

GHz), 16 GB DDR4 RAM, an NVIDIA RTX 2050 GPU (4 GB), and a 512 GB SSD, running Windows 11.

The malware detection pipeline was implemented in Python 3.9. Key libraries and frameworks included:

pefile for parsing PE headers and extracting imported functions.

Capstone disassembly engine for recovering additional API call references from executable code sections.

pandas for data handling and feature structuring.

scikit-learn for implementing classical machine learning classifiers and evaluation procedures.

TensorFlow/Keras for training and evaluating the ANN.

All random seeds were fixed across experiments to ensure reproducibility. Identical training or test splits were maintained for all models to guarantee fair comparison.

IV. Results and Discussion

A. Results

Thirteen classifiers were evaluated to identify the most effective algorithms for malware detection using shared API-call features. Table V summarizes the performance across four key metrics: Overall Accuracy (1), True Negative Rate (2), False Positive Rate (3), and Area Under the ROC Curve (4) [43]-[46].

The Extra Trees classifier achieved the highest overall accuracy (OA = 98.14%) and a strong AUC of 98.19%, confirming its suitability for handling high-dimensional, sparse binary feature vectors. The ANN achieved the highest AUC (99.50%), reflecting excellent discriminatory power between benign and malicious samples, although its OA was slightly lower at 97.67%.

Other strong performers included Logistic Regression (OA = 97.67%, AUC = 97.75%) and AdaBoost (OA = 97.67%, AUC = 97.69%). Gradient Boosting also performed competitively (OA = 97.21%, AUC = 97.31%), while Random Forest and LightGBM both reached OA = 96.74%. Traditional classifiers such as Naïve Bayes, LDA, and QDA underperformed, with OA below 85%, highlighting their limitations in modeling nonlinear feature interactions.

Notably, the Decision Tree classifier achieved an FPR of 0% (3), although its OA and AUC (95.35% and 95.61%) were lower than ensemble-based methods.

Table V shows the performance comparison of classifiers using FPR (3), TNR (2), OA (1), and AUC (4).

TABLE V Performance Comparison of Detectors

Algorithm	FPR (%)	TNR (%)	AUC (%)	OA (%)
Naïve Bayes	3.25	13.49	83.82	83.26
Logistic Regression	0.46	1.86	97.75	97.67
LDA	20	17.67	62.05	62.33
SVM	0.93	2.79	96.38	96.28
QDA	19.06	14.42	66.11	66.51
KNN	0.93	4.18	95.06	94.88
Decision Trees	0	4.65	95.61	95.35
ANN	0.93	1.39	99.50	97.67
Random Forest	0.46	2.79	96.87	96.74
Extra Trees	0.46	1.39	98.19	98.14
AdaBoost	0.93	1.39	97.69	97.67
Gradient Boosting	0.46	2.32	97.31	97.21
LightGBM	0.46	2.79	96.87	96.74

To evaluate the contribution of the proposed approach relative to prior studies, Table VI presents a comparative analysis. It contrasts the performance of the shared API-call-based Extra Trees classifier results with traditional string- and header-based approaches, as well as recent SOTA methods such as deep ensembles, hybrid static–dynamic frameworks, and attention-based models.

TABLE VI Comparative Analysis of the Proposed Method and Existing Malware Detection Approaches

Study	Features used	Model / Architecture	OA (%)	Interpretability	Complexity	Notes
Proposed Method (Extra Trees)	Shared API call vectors	Extra Trees Classifier	98.14	High	Low	Lightweight and fast; effective for sparse binary features
Balram, et al. [12]	String	Hybrid (LR/XGB)	98	Moderate	High	Highest accuracy; longer execution time
Baldangombo et al. [13]	PE Headers	J48	99	High	Moderate	Best standalone feature set; uses PCA
Azeez et al. [14]	PE Headers	Ensemble (7 NNs + ExtraTrees)	100	Low	Very High	Best performance; requires stacked ensemble training
Shijo and Salim [15]	Integrated (PSI + API 3/4-grams)	SVM	98.7	Low	Very High	Best accuracy; combines static/dynamic strengths
Yousuf et al. [16]	PE Header + PE Section (Integrated)	Random Forest, Stack Gen	99.5	Medium	Moderate-High	Uses IG/PCA for feature selection; combines static structural features.
Bensaoud and Kalita [17]	API Calls + Opcodes (8-grams)	CNN-LSTM-3	99.91	Medium	High	Combines CNN for spatial features and LSTM for sequential patterns.
Trizna et al. [18]	Dynamic API call traces	Self-Attention (Nebula)	99.8	Medium	High	SOTA dynamic analysis; resource-intensive
Seneviratne et al. [19]	Malware images (transformed binaries)	Vision Transformer (Self-Supervised)	97	Low	Very High	Requires GPU-heavy training; less interpretable

Mohammadian et al. [20]	Graph-based API dependency features	Graph Reduction + Explainable Learning	95.4	High	High	Improves explainability; complex preprocessing
Shokouhinejad et al. [21]	Graph-based features	Explainable Ensemble Learning	86.14	High	High	Balances interpretability with high accuracy
J. Tang [22]	Static + Dynamic fusion	Hybrid ML Ensemble	> 95	Medium	Very High	Strong accuracy but requires both runtime and static environments

The Receiver Operating Characteristic (ROC) curve for the Extra Trees classifier is shown in Fig. 3, illustrating its balance of high OA and AUC with low FPR.

Fig. 3 presents the ROC curve of the Extra Trees classifier, evaluated with OA (1), AUC (4), TNR (2), and FPR (3).

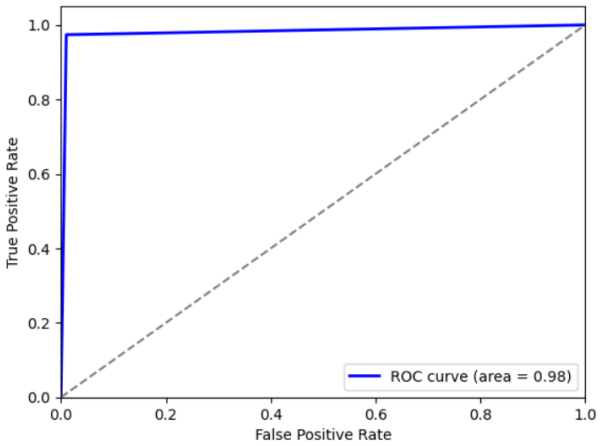


Fig. 3. ROC curve using extra trees classifier.

B. Discussion

The results confirm the effectiveness of tree-based ensemble models for static malware detection in sparse, binary feature spaces. The Extra Trees classifier offers the best balance between accuracy and interpretability, while the ANN model demonstrates superior AUC performance, suggesting that deep learning techniques can better capture subtle feature interactions.

In comparison with existing studies, the proposed framework offers several practical advantages, notably in terms of lightweight architecture, reduced feature redundancy, and improved computational efficiency. For instance, Azeez et al. [14] employed a complex ensemble of seven neural networks combined with Extra Trees to attain near-perfect detection accuracy, resulting in significant training overhead and deployment challenges. In contrast, the present approach circumvents such complexity, enhancing scalability and ease of integration. Furthermore, while Shijo and Salim [15] relied on sandboxed environments to extract dynamic behavioral features, the proposed static-only methodology eliminates the need for runtime execution, thereby simplifying implementation while maintaining competitive detection performance.

Balram et al. [12] and Baldangombo et al. [13] reported strong results using string- and header-based features but did not emphasize interpretability or feature reduction. Yousuf et al. [16] achieved an impressive 99.5% accuracy through feature fusion and advanced ensemble techniques; however, this came at the cost of increased system complexity.

More recent advances, such as self-attention models [18], [19] and explainable graph-based frameworks [20], [21], achieve high accuracy and improved interpretability but require significant computational resources, making them less suitable for lightweight or real-time deployments. Hybrid feature fusion approaches, such as that proposed by Tang [22], demonstrate the value of combining static and dynamic features but introduce additional integration challenges.

Overall, the proposed method achieves high accuracy (98.14% overall accuracy) and excellent discrimination (99.50% AUC) while using a compact and interpretable feature set of 558 shared API calls.

LIMITATIONS

As with most static approaches, the method may be less effective against heavily obfuscated or polymorphic malware and could face challenges related to concept drift, where emerging malware families exhibit behaviors not represented in the training data. Additionally, the moderate dataset size may limit generalization to large-scale, real-world scenarios.

PRACTICAL APPLICABILITY

Despite these limitations, the proposed framework is fast, lightweight, and interpretable, making it suitable for endpoint security systems and resource-constrained environments, where dynamic analysis is infeasible.

FUTURE WORK

Future research should focus on extending the analysis to larger and more imbalanced datasets would provide stronger evidence of generalizability. Moreover, integrating hybrid static–dynamic features could enhance resilience against obfuscation and adversarial evasion. Finally, exploring advanced architectures, such as CNN-LSTM hybrids or transformer-based models, may offer additional improvements in robustness and scalability.

V. CONCLUSION

This study presented a lightweight and interpretable malware detection framework that leverages shared API call behaviors extracted from benign and malicious PE files. Thirteen classifiers were systematically evaluated on a balanced dataset of 1,318 samples, demonstrating that high detection accuracy can be achieved with carefully selected static features. Among the evaluated models, the Extra Trees classifier achieved the best overall accuracy (98.14%), while maintaining high interpretability and computational efficiency. The ANN reached the highest AUC (99.50%), highlighting the ability of shallow deep learning models to capture nonlinear feature interactions. Overall, tree-based ensemble methods proved most effective for high-dimensional binary API-call features, offering a strong balance between detection power, efficiency, and interpretability.

In comparison with recent SOTA approaches such as hybrid CNN-LSTM models, graph-based learning, and transformer-based architectures, the proposed framework provides a more practical and resource-efficient solution. While those models often achieve higher accuracy on larger datasets, they typically introduce greater computational overhead and reduced interpretability. In contrast, the presented framework achieves competitive results while remaining lightweight, making it suitable for deployment in resource-constrained or real-time environments.

Despite these strengths, the study has several limitations. The exclusive reliance on static analysis may reduce robustness against highly obfuscated or adversarial malware, while the use of a moderately sized, balanced dataset limits generalizability to real-world malware distributions, which are often large-scale and highly imbalanced. Issues such as concept drift, adversarial evasion, and scalability in high-volume environments remain open challenges.

Future work should extend this research by validating the framework on larger and more diverse datasets, integrating dynamic and hybrid features, and investigating advanced ensemble or deep learning architectures such as CNN-LSTMs and transformer-based models. Additional directions include enhancing resilience against adversarial evasion techniques, addressing concept drift in evolving threat landscapes, and optimizing scalability for high-throughput detection scenarios. These improvements would strengthen both the theoretical and practical impact of the proposed approach, ensuring its continued relevance in modern malware detection.

REFERENCES

[1] Symantec Corporation, “Internet Security Threat Report,” Rep. 22, Mountain View, CA, USA, Apr. 2017. [Online]. Available: https://www.symantec.com/security-center

[2] S. Anderson, “Understanding the financial impact of cyber-attacks on businesses,” *J. Inf. Secur. Appl.*, vol. 40, pp. 133–142, Mar. 2018, doi: 10.1016/j.jisa.2018.02.009.

[3] W. Stallings and L. Brown, *Computer Security: Principles and Practice*, 4th ed. Boston, MA, USA: Pearson, 2018. [Online]. Available: https://www.pearsonhighered.com/assets/preface/0/1/3/2/0132775069.pdf

[4] M. Ahmad, “A study on ransomware attacks and their impact on the cybersecurity landscape,” *Int. J. Adv. Res. Comput. Sci.*, vol. 10, no. 4, pp. 285–292, Dec. 2019, doi: 10.26483/ijarcs.v10i4.6477.

[5] D. Harley and A. Lee, *Viruses Revealed*. New York, NY, USA: Osborne/McGraw-Hill, 2001. [Online]. Available: https://archive.org/details/virusesrevealed00harl

[6] R. A. Grimes, *Hacking the Hacker: Learn from the Experts Who Take Down Hackers*. Indianapolis, IN, USA: Wiley, 2017.

[7] IBM Security, *Cost of a Data Breach Report*, Armonk, NY, USA: IBM, 2024. [Online]. Available: https://www.ibm.com/reports/data-breach

[8] AV-TEST Institute, *Malware Statistics*, Magdeburg, Germany, 2024. [Online]. Available: https://www.av-test.org/en/statistics/malware/

[9] W. Tang *et al*., “Hybrid analysis framework for large-scale malware detection using machine learning,” *Comput. Secur.*, vol. 98, p. 102014, May 2020, doi: 10.1016/j.cose.2020.102014.

[10] A. Aghaei-Foroushani and M. Zincir-Heywood, “A comprehensive study of evasive malware techniques and their implications for current detection tools,” *Comput. Commun.*, vol. 145, pp. 241–260, Dec. 2019, doi: 10.1016/j.comcom.2019.06.014.

[11] S. Krishnan, Y. C. Tian, J. Sun, and H. Y. Liu, “Heuristic-based detection of code obfuscation malware,” *IEEE Access*, vol. 8, pp. 7198–7207, Dec. 2020, doi: 10.1109/ACCESS.2020.2972442.

[12] N. Balram, G. Hsieh, and C. McFall, “Static malware analysis using machine learning algorithms on APT1 dataset with string and PE header features,” in *Proc. Int. Conf. Comput. Sci. Comput. Intell. (CSCI)*, Las Vegas, NV, USA, Dec. 2019, pp. 90–95, doi: 10.1109/CSCI49370.2019.00022.

[13] U. Baldangombo, N. Jambaljav, and S.-J. Horng, “A static malware detection system using data mining methods,” *arXiv*, arXiv:1308.2831, Aug. 2013. [Online]. Available: http://arxiv.org/abs/1308.2831

[14] N. A. Azeez, O. E. Odufuwa, S. Misra, J. Oluranti, and R. Damasevicius, “Windows PE malware detection using ensemble learning,” *Appl. Sci.*, vol. 11, no. 4, Art. no. 1477, 2021, doi: 10.3390/app11041477.

[15] P. V. Shijo and A. Salim, “Integrated static and dynamic analysis for malware detection,” *Procedia Comput. Sci.*, vol. 46, pp. 804–811, 2015, doi: 10.1016/j.procs.2015.02.149.

[16] M. I. Yousuf, I. Anwer, A. Riasat, K. T. Zia, and S. Kim, “Windows malware detection based on static analysis with multiple features,” *PeerJ Comput. Sci.*, vol. 9, p. e1319, Apr. 2023, doi: 10.7717/peerj-cs.1319.

[17] A. Bensaoud and J. Kalita, “CNN-LSTM and transfer learning models for malware classification based on opcodes and API calls,” *Knowl.-Based Syst.*, vol. 290, p. 111543, Apr. 2024, doi: 10.1016/j.knosys.2024.111543.

[18] D. Trizna, “Nebula: Self-attention for dynamic malware analysis,” arXiv, vol. abs/2310.10664, Oct. 2023. [Online]. Available: https://arxiv.org/abs/2310.10664

[19] S. Seneviratne, R. Shariffdeen, S. Rasnayaka, and N. Kasthuriarachchi, “Self-supervised vision transformers for malware detection,” arXiv, vol. abs/2208.07049, Aug. 2022. [Online]. Available: https://arxiv.org/abs/2208.07049

[20] H. Mohammadian, G. Higgins, S. Ansong, R. Razavi-Far, and A. A. Ghorbani, “Explainable malware detection through integrated graph reduction and learning techniques,” arXiv, vol. abs/2412.03634, Dec. 2024. [Online]. Available: https://arxiv.org/abs/2412.03634

[21] H. Shokouhinejad, R. Razavi-Far, G. Higgins, and A. A. Ghorbani, “Explainable ensemble learning for graph-based malware detection,” arXiv, vol. abs/2508.09801, Aug. 2025. [Online]. Available: https://arxiv.org/abs/2508.09801

[22] J. Tang, “Fusion of static and dynamic features for malware detection,” Proc. ACE Conf., pp. 1–8, Jul. 2025. [Online]. Available: https://direct.ewa.pub/proceedings/ace/article/view/24689

[23] P. Dong, Z. Wang, and J. Cheng, “A new static malware detection method using features of PE headers,” *J. Comput. Syst. Sci.*, vol. 89, pp. 449–464, May 2017, doi: 10.1016/j.jcss.2017.05.001.

[24] F. Schuster *et al*., “Malware Zoo: A dataset of executable and online malware samples for academic and industrial research,” in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS)*, pp. 781–796, 2017, doi: 10.1145/3133956.3134073.

[25] VirusShare, “Comprehensive Repository of Malware Samples.” [Online]. Available: https://virusshare.com

[26] F. Rafique and M. K. Luhach, “PE file format features for malware detection: A systematic review,” *IEEE Access*, vol. 10, pp. 3035–3044, Jan. 2022, doi: 10.1109/ACCESS.2022.3140032.

[27] A. Mosse, J. Howard, and G. Hill, “Analysis of API calls in portable executable (PE) files for threat analysis,” in *Proc. ACM SIGSAC Conf. Artif. Intell. Cybersecurity*, pp. 55–64, 2019.

[28] E. Andriesse, X. Fratantonio, A. Sasse, and C. Kruegel, “Practical binary code analysis using Capstone,” in *Proc. IEEE Secur. Privacy Workshops (SPW)*, pp. 1–7, 2018, doi: 10.1109/SPW.2018.00010.

[29] V. Talib and S. Girkar, “Detection of malware through API call sequence and classification techniques,” *J. Netw. Comput. Appl.*, vol. 50, pp. 129–143, Jun. 2018, doi: 10.1016/j.jnca.2017.03.015.

[30] G. D'Angelo and R. Di Pietro, "Unsupervised detection of malware via dynamic and static disassembly analysis," in *Proc. IEEE TrustCom*, pp. 91–99, Aug. 2016, doi: 10.1109/TrustCom.2016.016.

[31] Capstone, "Capstone: Lightweight Multi-Architecture Disassembly Framework." [Online]. Available: https://www.capstone-engine.org

[32] M. Saud, U. Mahbub, and A. Anwar, "Analysis of portable executable (PE) file metadata for malware classification using static analysis," in *Proc. ACM SIGSAC Conf. Artif. Intell. Cybersecurity*, pp. 104–113, 2019, doi: 10.1145/3313407.3314009.

[33] S. Kariyappa and H. Ambre, "Static malware analysis with PE parsing: Exploring Python libraries like pefile for threat modeling," *Int. J. Adv. Comput. Sci. Appl. (IJACSA)*, vol. 10, no. 4, pp. 45–54, Apr. 2019, doi: 10.14569/IJACSA.2019.010042.

[34] P. Sun and Y. Wang, "Binary malware classification using supervised machine learning and API feature analysis," in *Proc. ACM Int. Conf. Inf. Knowl. Manage. (CIKM)*, pp. 1555–1562, 2017, doi: 10.1145/3132847.3133071.

[35] F. Santos, C. G. Comin, and A. Capurro, "Static analysis of Windows malware using Capstone disassembly for API call extraction," *IEEE Access*, vol. 8, pp. 6504–6517, Jan. 2020, doi: 10.1109/ACCESS.2019.2963872.

[36] W. Tang, S. Yan, and Z. Yan, "Hybrid data preprocessing for robust malware detection using API features," *IEEE Access*, vol. 9, pp. 34177–34187, Mar. 2021, doi: 10.1109/ACCESS.2021.3060497.

[37] D. Borkin, A. Némethová, G. Michaľčonok, and K. Maiorov, "Impact of data normalization on classification model accuracy," Res. Pap. Fac. Mater. Sci. Technol. Slovak Univ. Technol., vol. 27, no. 45, pp. 79–84, Sep. 2019, doi: 10.2478/rput-2019-0029.

[38] D. Mohaisen and O. Alrawi, "AMAL: High-level comprehensive malware analysis and data preprocessing framework," in *Proc. ACM SIGSAC Symp. Inf., Comput. Commun. Secur. (Asia CCS)*, pp. 101–111, Jun. 2014.

[39] P. C. Zhu and Y. X. Zhou, "Practical data augmentation and cleaning for malware analysis," *J. Comput. Theor. Nanosci.*, vol. 14, no. 1, pp. 134–141, Jan. 2017, doi: 10.1166/jctn.2017.6861.

[40] D. Kong and G. Yan, "Discerning malware program structure through API sequence similarity," in *Proc. IEEE Secur. Privacy Symp.*, pp. 512–522, May 2013, doi: 10.1109/SP.2013.40.

[41] N. Japkowicz and M. Shah, *Evaluating Learning Algorithms: A Classification Perspective*. Cambridge, U.K.: Cambridge Univ. Press, 2011.

[42] F. Schuster and G. Jacob, "Machine learning-based data cleaning in malware detection pipelines," *Comput. Secur.*, vol. 70, pp. 79–87, Sep. 2017, doi: 10.1016/j.cose.2017.06.009.

[43] P. Vemuri, B. Mukherjee, and M. M. S. Poonkodi, "Comparative analysis of machine learning classifiers for malware detection," *IEEE Trans. Emerg. Topics Comput.*, vol. 5, no. 4, pp. 476–489, Dec. 2017, doi: 10.1109/TETC.2016.2602621.

[44] H. Zhang, W. Dai, and Y. Xu, "Malware detection using random forest and decision tree classifiers," *Comput. Secur.*, vol. 60, pp. 23–31, Oct. 2016, doi: 10.1016/j.cose.2016.04.004.

[45] J. Brownlee, *Master Machine Learning Algorithms: Discover How They Work and Implement Them from Scratch*, 2nd ed. Victoria, Australia: Machine Learning Mastery, 2019.

[46] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, pp. 785–794, Aug. 2016, doi: 10.1145/2939672.2939785.

Intrusion Detection Using Machine Learning and Deep Learning

Fatima Jobran ALzaher, Asma AlJarullah
Department of Informatics and Computer Systems, King Khalid University, 61421, Alfara, Abha, Saudi Arabia

***Abstract*—As cyberattacks grow in prevalence, Intrusion Detection Systems (IDS) have become critical for securing network infrastructures. This study proposes an efficient IDS framework utilizing both machine learning (ML) and deep learning (DL) algorithms. The framework is evaluated on the "NF-UNSW-NB15-v2" dataset, which comprises a blend of normal and malicious traffic. A diverse set of advanced models—including Deep Neural Networks (DNN), Long Short-Term Memory (LSTM) networks, eXtreme Gradient Boosting (XGBoost), Random Forest (RF), and K-Nearest Neighbors (KNN)—is deployed for intrusion detection. The approach encompasses both binary classification (normal vs. malicious) and multi-class classification (specific attack categories). Preprocessing steps include feature standardization using StandardScaler, class imbalance correction via SMOTE, and dimensionality reduction through Principal Component Analysis (PCA). Results show that Random Forest and XGBoost models achieve high accuracy in binary classification with F1-scores approaching 0.97, while XGBoost attains the best macro F1-score (0.71) in multi-class tasks. Additionally, RF and XGBoost demonstrate the fastest inference times, underscoring their suitability for real-time deployment. This work contributes a scalable and optimized IDS pipeline for enhancing cybersecurity resilience.**

***Keywords*—Cybersecurity; cyber-attack; intrusion detection system; machine learning; deep learning**

I. Introduction

In recent years, the cyber world witnessed the most phenomenal increase ever of cyber threats that targeted individuals, businesses, and governments. The cost of cybercrime at the global level is anticipated at $9.5 trillion by 2024, which amounts to $26 billion per day or $18 million per minute [1]. The growing rate of cybercrime emphasizes the need for proper cybersecurity controls.

Small and medium-sized businesses are most vulnerable with 69% of these experiencing at least one cyber-attack within the past year [1]. Yet 80% of the SMBs are largely unimplemented with the utilization of Privileged Access Management solutions while fewer than 60% of the enterprises are using vital cybersecurity practices of password managers, two-factor authentication, and cybersecurity education [1]. The shift toward remote work amplified the fear of security since 72% of businesses are concerned about the danger it presents and 80% of cybersecurity professionals confirm that the danger increased since 2020 [1].

Phishing and ransomware attacks grew more complex and more numerous. Security professionals saw 62% more phishing assaults within the recent years with 79% of account takeover attacks originating from the type of phishing attack that occurs with the use of phish emails [1]. The ransomware attacks grew with 70% of the attacks focusing on the small business market and the number of ransomware teams actively present more than doubling year over year [1].

Adversaries now leverage automation and AI to accelerate reconnaissance, weaponization, and evasion, fueling an arms race that compels defenders to adopt more adaptive, data-driven countermeasures [2].

Traditional intrusion detection systems (IDS) that rely on static signatures or hand-crafted rules struggle with previously unseen or rapidly morphing threats [3]. Machine learning (ML) and artificial intelligence (AI) enable dynamic threat modeling, anomaly detection, and behavioral analysis, allowing IDSs to generalize to novel attacks while reducing false positives [3].

Despite notable advancements in IDS research, a critical limitation persists across most studies: the insufficient handling of class imbalance, which leads to poor detection rates for minority attack types. Previous works have primarily focused on maximizing overall accuracy, often at the expense of rare class detection, resulting in inflated performance metrics dominated by frequent categories. Furthermore, inconsistencies in preprocessing pipelines, a lack of standardized evaluation on modern NetFlow-based datasets like NF-UNSW-NB15-v2, and the limited integration of a diverse set of both machine and deep learning techniques within a single, optimized framework limit the robustness and practical applicability of existing IDS solutions. This study aims to bridge these gaps by proposing an integrated pipeline that combines SMOTE-based class balancing, PCA-driven feature reduction, and a comprehensive evaluation of multiple ML and DL models to enhance minority class detection and overall performance on the NF-UNSW-NB15-v2 dataset.

The remainder of the paper is organized as follows. Section II reviews related work on ML/DL-based IDS. Section III details the methodology, including the dataset, preprocessing (standardization, imbalance handling, dimensionality reduction, and train–test split), model specifications, and evaluation metrics. Section IV reports the experimental results for both binary and multi-class settings and examines inference-time performance for real-time applicability. Section V discusses key findings, practical implications, and limitations. Section VI concludes and outlines directions for future research.

II. Related Work

Kasongo and Sun [4] conducted a performance analysis of Intrusion Detection Systems (IDS) by applying a feature

selection algorithm to the UNSW-NB15 dataset. They emphasized the importance of reducing feature dimensionality to enhance IDS accuracy. The study used five machine learning classifiers—SVM, KNN, Logistic Regression, Artificial Neural Networks (ANN), and Decision Trees (DT)—and showed that XGBoost-based feature selection significantly improved model performance, especially for DTs. However, the study did not address class imbalance, which led to lower F1-scores for minority classes.

Kumar et al. [5] proposed an integrated rule-based IDS using both the UNSW-NB15 and RTNITP18 datasets. Their approach employed Decision Tree classifiers (C5, CHAID, CART, and QUEST) to detect five attack types: Generic, Probe, DoS, Exploit, and Normal. The study reported improved accuracy and reduced false alarm rates through rule-based modeling and information gain-based feature selection. Nonetheless, it lacked proper handling of class imbalance and yielded low precision and recall for critical attack types.

More et al. [6] compared several supervised ML techniques for identifying deceitful emails with filtering approaches and through use of the WEKA toolset. The work identifies a weakness in conventional Bayesian filtering, effective in spam filtering but not in high false positive cases. To counteract, several classifiers, such as RF and SVM, have been incorporated and utilized for enhancing accuracy and minimizing false positives in classification. In its use, feature extraction via Naïve Bayes and an evaluation tool developed in WEKA facilitated testing of numerous algorithms for classification in a thorough manner. Experimental tests showed that RF and SVM performed better in enhancing positive and negative actual values and overall accuracy over 96%. The work identifies the use of hybrid classification approaches in improving deceitful message detection and minimizing security threats posed through spammers.

Tahri et al. [7] have designed an IDS with ML algorithms for enhancing network security. As communications through electronic means have increased, IDS proves to be a useful tool for discovering hostile activity in network communications. In the current work, three classifiers, Naïve Bayes, SVM, and KNN, have been compared for performance with two benchmark datasets, namely, “NSL-KDD” and “UNSW-NB15”. In part one of work, three classifiers have been compared with the use of “UNSW-NB15”, and then for a proper analysis, best-performing algorithm is utilized for testing with “NSL-KDD”. As per work, SVM outperforms all classifiers in terms of accuracy consistently, with 97.77% accuracy for “UNSW-NB15” and 97.29% accuracy for NSL-KDD. In conclusion, SVM proves to be an effective intrusion detection classifier, and future work will attempt to make its processing efficient and integrate it in real-time security tools such as a firewall.

Musa et al. [8] review the application of ML algorithms in IDS for enhancing network security through observation of traffic and intrusion activity, and IDS is distinguished between anomaly-based and signature-based detection, with the first identifying abnormalities in behavior and the second employing predefined attack signatures. Various types of ML approaches, including single, hybrid, and ensemble classifiers, are contrasted and compared over seven datasets, with the consequence that single classifiers fall below both ensemble and hybrid classifiers in terms of accuracy and detection performance. Comparison between algorithms including SVM, DTs, RF, and Neural Networks identifies that ensemble approaches, including stacking classifiers, have a significant impact in intrusion detection improvement. Challenges include feature selection improvement, testing over a range of and updated datasets, and minimizing false positive values. Optimizing hybrid models, minimizing computational overload, and enhancing real-time intrusion capabilities have been suggested for future work, according to the authors.

Samantaray et al. in [9] conducted a comparative study on ML model implementation in intrusion detection in IoT-based networks. The research is centered on increasing threats in the security of IoT networks and the need to utilize efficient IDS in order to mitigate them. The research uses the “UNSW-NB15 (DS-1)” and “NF-UNSW-NB15 (DS-2)” datasets in comparing models based on ML like SVM, KNN, Logistic Regression, Naïve Bayes, DT, and RF. The feature scaling is based on a method involving the usage of the MaxAbsScaler algorithm in order to increase efficiency in classification. The results highlight the usage of the RF classifier in achieving the highest precision in generating the most accurate outcome with a gain in the rate of detection from 60% to 94% in the DS-2 dataset. The research focuses on efficiency in ML usage in intrusion detection and supports future research on implementation with improved feature selection and DL.

Sayed et al. in [10] conducted research with a focus on optimizing the efficiency of DNN-driven IoT intrusion detection systems (IDS). Because the IoT devices are under threat and there is a limitation in the process ability and in features in security, the researchers provided two CNN models, namely IoTCNN and MyCNN, with a purpose to classify intrusion in the network. The "NF-UNSW-NB15-v2" dataset was used in the research, and the stream network data was converted into RGB images in order to train the models. Results indicated the efficiency of the models in the detection of various intrusion types, and in the majority of intrusion categories, the precision of the models improved. The research confirms the efficiency of anomaly-based IoT security based on DL and calls for improvement in the handling of imbalances in the class and the optimization of the hyperparameters.

Table I provides a structured summary of key related studies, highlighting the datasets, methodologies, and performance metrics used, which helps position the present work within the broader landscape of IDS research.

Despite notable advancements in IDS research, a recurring limitation across most studies is the insufficient handling of class imbalance and the resulting poor detection of minority attack types. Previous works primarily focused on improving overall accuracy without explicitly addressing the critical challenge of rare class detection, often leading to inflated performance metrics dominated by frequent attack categories. In addition, inconsistencies in preprocessing, lack of standardized evaluation on newer datasets, and limited integration of deep learning techniques further limit the robustness of existing IDS solutions. Building on these gaps, this research proposes an

integrated approach combining SMOTE-based class balancing, PCA-driven feature selection, ML/DL modeling to enhance minority class detection and overall intrusion detection performance on the NF-UNSW-NB15-v2 dataset [20].

TABLE I. SUMMARY OF RELATED WORK

Year	Research Title	Dataset	Machine Learning / Deep Learning Model(s)	Preprocessing	Classification Type	Performance Metrics	Findings	Limitations	Ref
2020	Performance analysis of IDS using a feature selection method on UNSW-NB15	UNSW-NB15 [21]	SVM, KNN, Logistic Regression, ANN, Decision Tree	Min-Max normalization; Feature selection with XGBoost	Binary and Multi-class	Binary: Accuracy 90.85%, Precision 80.33%, Recall 98.38%, F1-score 88.45% Multi-class: Accuracy 77.51%, Precision 79.50%, Recall 77.53%, F1-score 77.28%	Feature selection (XGBoost) improved ML model performance.	No class imbalance handling; poor F1-scores for minority classes.	[4]
2020	An integrated rule-based intrusion detection system on UNSW-NB15 and RTNITP18	UNSW-NB15 [21], RTNITP18 [5]	Decision Trees (C5, CHAID, CART, QUEST)	Feature selection using Information Gain; K-Means clustering	Multi-class	Accuracy 84.83%, Approximate F1-score 68.13%	Rule-based modeling reduced false alarms in IDS.	No class imbalance handling; low precision and recall for critical attacks.	[5]
2015	Evaluation of deceptive mails using filtering & WEKA	SpamBase [22], Ling-Spam [23], Enron [24], PU1 [25], PU2 [25]	Random Forest, SVM, Naïve Bayes	Tokenization; Feature extraction; Term Frequency normalization	Binary	Accuracy: RF 98.9%, SVM 98.4%, NB 93.2%	Ensemble models (Random Forest) achieved high classification accuracy.	No class imbalance handling; no minority class evaluation.	[6]
2022	Intrusion Detection System using machine learning algorithms	UNSW-NB15 [21], NSL-KDD [26]	SVM, KNN, Naïve Bayes	Feature selection using mutual information;	Binary and Multi-class	Binary: SVM Accuracy 97.78% Multi-class: SVM Accuracy 97.29%	SVM achieved high accuracy on IDS datasets without heavy feature engineering.	No class imbalance handling; no minority class evaluation.	[7]
2020	Review of machine learning techniques for IDS across different datasets	KDDCup'99 [27], NSL-KDD [26], Kyoto2006+ [28], AWID [29], CIC-IDS2017 [30], UNSW-NB15 [21], UGR'16 [31]	SVM, Random Forest, Decision Tree, KNN, ANN, XGBoost, AdaBoost	Dataset-specific feature engineering; Standard normalization or scaling where needed	Binary and Multi-class	Binary: Ensemble models achieved >99% Accuracy Multi-class: Accuracy ~0.99, Macro F1 ~0.89	Ensemble methods (e.g., XGBoost) consistently outperformed individual classifiers.	No class imbalance handling; no minority class evaluation.	[8]
2024	Comparative assessment of ML algorithms in IoT-based network intrusion detection	UNSW-NB15 (DS-1), NF-UNSW-NB15 (DS-2) [20]	SVM, KNN, Logistic Regression, Naïve Bayes, Random Forest	MaxAbsScaler normalization	Multi-class	Accuracy: RF 60% (DS-1), 94% (DS-2)	Feature scaling (MaxAbsScaler) improved IDS model stability.	No class imbalance handling; no minority class evaluation.	[9]
2022	Augmenting IoT intrusion detection system performance using deep neural networks	NF-UNSW-NB15-v2 [20]	CNN	FFT-based NetFlow transformation to RGB images; Image normalization	Multi-class	Accuracy ~99% (frequent classes); poor F1-scores for minority classes	CNNs using NetFlow-to-image transformation achieved high accuracy for frequent attacks.	No class imbalance handling; poor F1-scores for minority classes.	[10]

III. METHODOLOGY

The methodology for this project is structured into several essential steps, as depicted in Fig. 1.

The proposed method for intrusion detection using deep learning (DL) and machine learning (ML) follows a structured pipeline:

1) Dataset: The NF-UNSW-NB15 dataset is used, comprising a combination of normal and malicious network traffic. It provides a realistic foundation for evaluating intrusion detection models.

Preprocessing Steps:

- Standardization: Normalizes the data to have the same feature scaling.
- PCA (Principal Component Analysis): Reduces dimensionality in a way that maximizes computational efficiency while retaining significant features.
- Oversampling: Balances the dataset and treats imbalances in classes, optimizing performance on minority attack classes.

2) Model training: Preprocessed data is fed to ML (e.g., XGBoost, RF, KNN) and DL (e.g., DNN, LSTM) models for training.

3) Evaluation metrics: Model performance is evaluated using conventional metrics, including accuracy, precision, recall, F1-score, and AUC-ROC, to ensure robust and comprehensive assessment of intrusion detection effectiveness.

B. Dataset

The NetFlow-based variant of the UNSW-NB15 dataset, referred to as NF-UNSW-NB15, incorporates additional flow-level features and is labeled according to specific attack categories. The original dataset comprises 2,390,275 network flow records, including 95,053 attacks (3.98%) and 2,295,222 benign flows (96.02%). These attack records are further divided into nine subtypes, as summarized in Table II [19]. For this study, the dataset was obtained from Kaggle, where the official version was uploaded by the author after removing duplicate rows, reducing the total count to 1,986,745. Therefore, the dataset used in this research is consistent with the original release, except for the exclusion of duplicates to improve data integrity and processing efficiency.

Mohanad Sarhan et al. [20] proposed a standardized feature set for network intrusion detection datasets to improve detection performance through the application of machine learning techniques. Their approach leverages NetFlow v9 features, which are widely supported by network devices and proven to be effective for traffic analysis. The proposed feature set includes 43 numerical, flow-based attributes designed to facilitate accurate and consistent detection of security events. By promoting dataset standardization, this feature set simplifies model evaluation, enhances compatibility for dataset merging, and supports real-world deployment of intrusion detection systems.

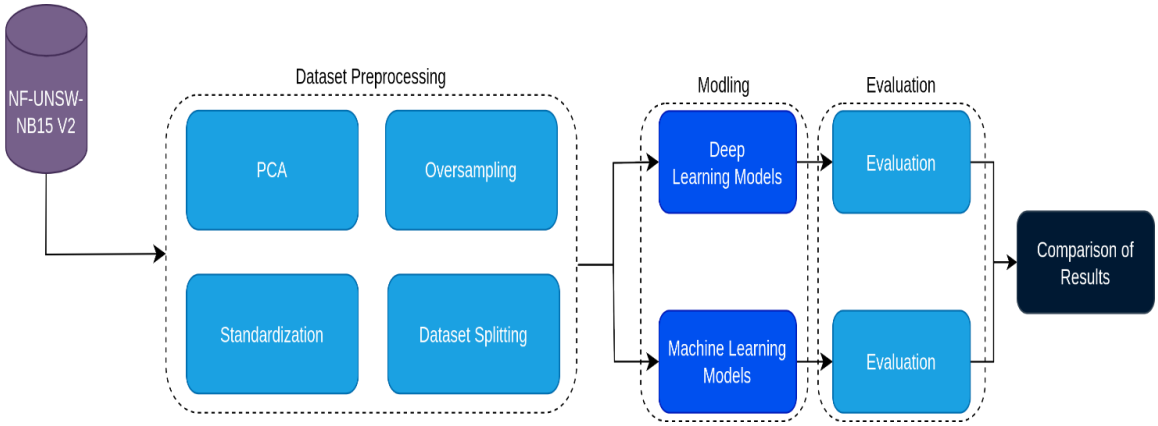


Fig. 1. Proposed method.

TABLE II. ATTACK TYPES IN "NF-UNSW-NB15"

Class	Count	Description
Benign	2295222	Normal, non-malicious network traffic.
Fuzzers	22310	An attack where large volumes of random data are sent to a system to cause crashes and identify security vulnerabilities.
Analysis	2299	A category of threats targeting web applications through ports, emails, and scripts.
Backdoor	2169	A method that bypasses security mechanisms by responding to specially crafted client requests.
DoS	5794	An attack that overwhelms a system's resources to disrupt access to its data or services.
Exploits	31551	Sequences of commands used to manipulate a system by exploiting known vulnerabilities.
Generic	16560	A cryptographic attack that causes collisions in block cipher encryption.
Reconnaissance	12779	Also known as probing, this technique involves gathering information about a network host.
Shellcode	1427	Malicious code designed to take control of a victim's system.
Worms	164	Self-replicating attacks that spread across multiple computers.

TABLE III. FEATURES IN "NF-UNSW-NB15"

	Field Name	Data Type	Description
1	IPV4_SRC_ADDR	String	Source IPv4 address
2	IPV4_DST_ADDR	String	Destination IPv4 address
3	L4_SRC_PORT	Integer	Source port number for IPv4
4	L4_DST_PORT	Integer	Destination port number for IPv4
5	PROTOCOL	Integer	Byte value representing the IP protocol identifier
6	L7_PROTO	Integer	Numeric identifier for the Layer 7 protocol
7	IN_BYTES	Integer	Total incoming bytes
8	OUT_BYTES	Integer	Total outgoing bytes
9	IN_PKTS	Integer	Count of incoming packets
10	OUT_PKTS	Integer	Count of outgoing packets
11	FLOW_DURATION_MILLISECONDS	Float	Duration of the flow in milliseconds
12	TCP_FLAGS	Integer	Aggregated TCP flags
13	CLIENT_TCP_FLAGS	Integer	Aggregated TCP flags from the client side
14	SERVER_TCP_FLAGS	Integer	Aggregated TCP flags from the server side
15	DURATION_IN	Float	Duration of the client-to-server stream (in milliseconds)
16	DURATION_OUT	Float	Duration of the server-to-client stream (in milliseconds)
17	MIN_TTL	Integer	Minimum Time-to-Live (TTL) value observed in the flow
18	MAX_TTL	Integer	Maximum Time-to-Live (TTL) value observed in the flow
19	LONGEST_FLOW_PKT	Integer	Size (in bytes) of the longest packet in the flow
20	SHORTEST_FLOW_PKT	Integer	Size (in bytes) of the shortest packet in the flow
21	MIN_IP_PKT_LEN	Integer	Length of the smallest observed IP packet in the flow
22	MAX_IP_PKT_LEN	Integer	Length of the largest observed IP packet in the flow
23	SRC_TO_DST_SECOND_BYTES	Float	Rate of bytes sent from source to destination (bytes per second)
24	DST_TO_SRC_SECOND_BYTES	Float	Rate of bytes sent from destination to source (bytes per second)
25	RETRANSMITTED_IN_BYTES	Integer	Count of retransmitted TCP bytes from source to destination
26	RETRANSMITTED_IN_PKTS	Integer	Count of retransmitted TCP packets from source to destination
27	RETRANSMITTED_OUT_BYTES	Integer	Count of retransmitted TCP bytes from destination to source
28	RETRANSMITTED_OUT_PKTS	Integer	Count of retransmitted TCP packets from destination to source
29	SRC_TO_DST_AVG_THROUGHPUT	Float	Average throughput (bps) from source to destination
30	DST_TO_SRC_AVG_THROUGHPUT	Float	Average throughput (bps) from destination to source
31	NUM_PKTS_UP_TO_128_BYTES	Integer	Number of packets with an IP size of 128 bytes or less
32	NUM_PKTS_128_TO_256_BYTES	Integer	Number of packets with an IP size between 128 and 256 bytes
33	NUM_PKTS_256_TO_512_BYTES	Integer	Number of packets with an IP size between 256 and 512 bytes
34	NUM_PKTS_512_TO_1024_BYTES	Integer	Number of packets with an IP size between 512 and 1024 bytes
35	NUM_PKTS_1024_TO_1514_BYTES	Integer	Number of packets with an IP size between 1024 and 1514 bytes
36	TCP_WIN_MAX_IN	Integer	Maximum TCP window size from source to destination
37	TCP_WIN_MAX_OUT	Integer	Maximum TCP window size from destination to source
38	ICMP_TYPE	Integer	ICMP type combined with ICMP code (ICMP Type * 256 + ICMP Code)
39	ICMP_IPV4_TYPE	Integer	ICMP type identifier for IPv4
40	DNS_QUERY_ID	Integer	Transaction ID of a DNS query
41	DNS_QUERY_TYPE	Integer	Type of DNS query (e.g., 1 = A, 2 = NS, etc.)
42	DNS_TTL_ANSWER	Integer	Time-to-Live (TTL) value of the first A record, if available
43	FTP_COMMAND_RET_CODE	Integer	Return code for an FTP client command

Table III lists the feature set of the NF-UNSW-NB15 dataset, detailing the flow characteristics captured for intrusion detection analysis.

Fig. 2 illustrates the significant class imbalance in the dataset, with benign (normal) traffic overwhelmingly dominating all categories of malicious traffic. Among the attack types, "Exploits" and "Fuzzers" appear most frequently, while others such as "Reconnaissance," "DoS," and especially "Worms" occur far less often. This imbalance poses challenges for accurate model training and may necessitate the use of resampling methods or advanced techniques to improve detection performance, particularly for minority attack classes.

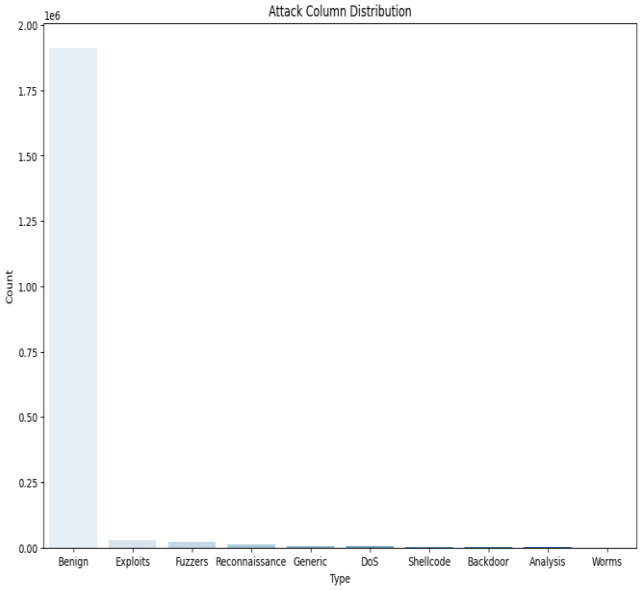


Fig. 2. Attack column distribution.

Fig. 3 displays how normal and malicious flows are distributed in the dataset, revealing a pronounced class imbalance: legitimate traffic vastly outweighs attack traffic.

Such skew can hamper model effectiveness, so remedies like SMOTE oversampling or class-weight adjustment are advisable to achieve more reliable classification.

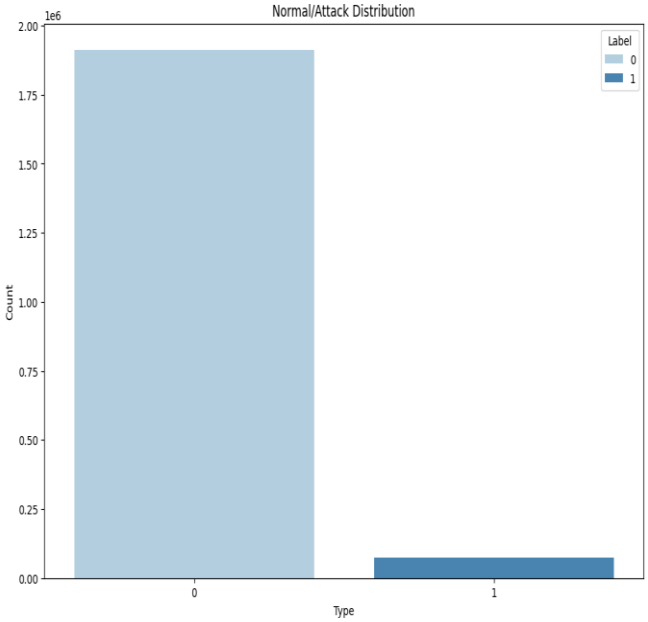


Fig. 3. Label column distribution.

Fig. 4 shows the distribution of different types of attacks in the dataset. "Exploits" is the highest frequency type, with "Fuzzers" and "Reconnaissance" following, and "Worms" and "Analysis" are the least frequent. The imbalanced distribution among attack types implies the need for careful model training in order to correctly identify in every category.

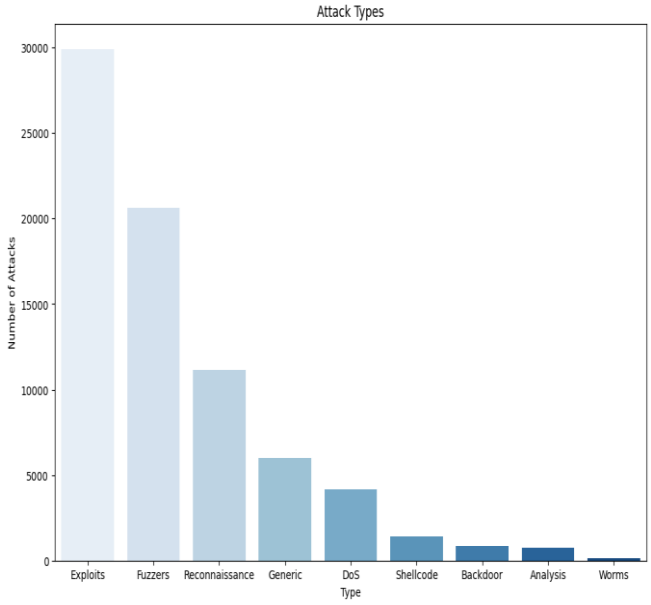


Fig. 4. Count of every attack type.

C. Dataset Preprocessing

To ensure high-quality input for model training, the NF-UNSW-NB15-v2 dataset undergoes a series of preprocessing steps designed to enhance learning efficiency and model performance.

1) Standardization: Standardization was applied using the StandardScaler, which transformed each numerical feature to have a mean of zero and a standard deviation of one. This normalization process ensured that all features contributed proportionally to the learning process, preventing any single feature with a large magnitude from disproportionately influencing the model. Standardization was particularly important for algorithms sensitive to feature scale, such as K-Nearest Neighbors (KNN) and Deep Neural Networks (DNN), which rely on distance-based calculations and gradient-based optimization, respectively. This approach follows best practices outlined in prior literature and common implementations such as scikit-learn [11].

2) Handling class imbalance: To address the class imbalance issue in the dataset, the Synthetic Minority Over-sampling Technique (SMOTE) [32] was applied exclusively to the training set after splitting the data into 70% for training and 30% for testing. This approach was intentionally adopted to prevent data leakage and to ensure an unbiased evaluation of the model's performance on unseen data.

SMOTE generates synthetic samples for minority classes by interpolating between existing instances rather than simply duplicating them. For each minority sample in the training set, the algorithm identifies its k-nearest neighbors (commonly k=5) within the same class. It then selects one neighbor at random and creates a new instance by generating a point along the feature-space line segment connecting the original sample and its neighbor. This synthetic instance inherits the statistical

characteristics of both points, resulting in a new example that is both realistic and non-redundant [32].

By enriching the training set with these synthetic examples, the class distribution becomes more balanced. This allows learning algorithms to be exposed to a wider variety of patterns and variations within the minority classes, which enhances their ability to generalize and improves detection of rare attack types. The test set was left untouched to preserve the original distribution and maintain the integrity of the evaluation.

3) Dimensionality reduction: To address the high dimensionality of the dataset, which consists of 43 flow-based NetFlow features, Principal Component Analysis (PCA) was employed as an effective dimensionality reduction technique. PCA transforms the original feature set into a smaller number of uncorrelated components while retaining the majority of the data's variance. This transformation reduces computational complexity, accelerates model training, and mitigates the risk of overfitting by eliminating redundant or less informative attributes [13].

An initial correlation analysis of the raw NetFlow features revealed strong linear relationships among several variables. For instance, IN_PKTS and OUT_PKTS exhibited a Pearson correlation coefficient of approximately 0.99, while FLOW_DURATION_MILLISECONDS was highly correlated with both DURATION_IN and DURATION_OUT. Similarly, MIN_TTL and MAX_TTL showed correlations exceeding 0.90. These relationships, visualized in Fig. 5, support the application of PCA to reduce multicollinearity and noise within the dataset.

To validate the independence of the PCA components, a correlation matrix was generated to examine the relationships between the extracted components. As shown in Fig. 6, the components are effectively uncorrelated, demonstrating that PCA successfully transforms the original feature space into a set of orthogonal, linearly independent dimensions. This orthogonality reinforces PCA's suitability for improving model robustness and reducing feature redundancy.

Overall, PCA proved especially beneficial for ensemble models such as Random Forest and XGBoost, which can be negatively affected by irrelevant or highly correlated features. Its use enhanced model focus, reduced overfitting, and improved interpretability within the intrusion detection pipeline.

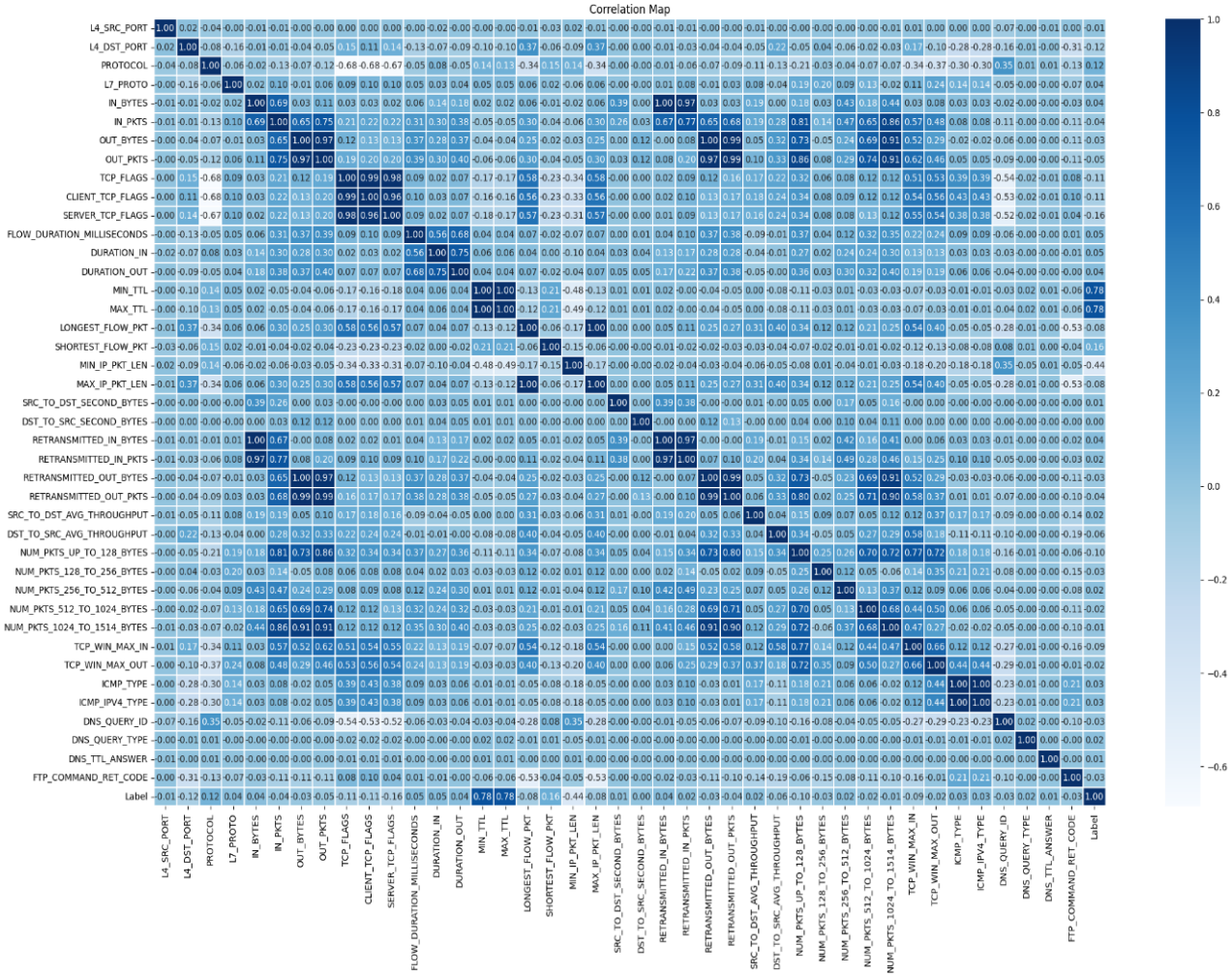


Fig. 5. Correlation heatmap of original NetFlow features.

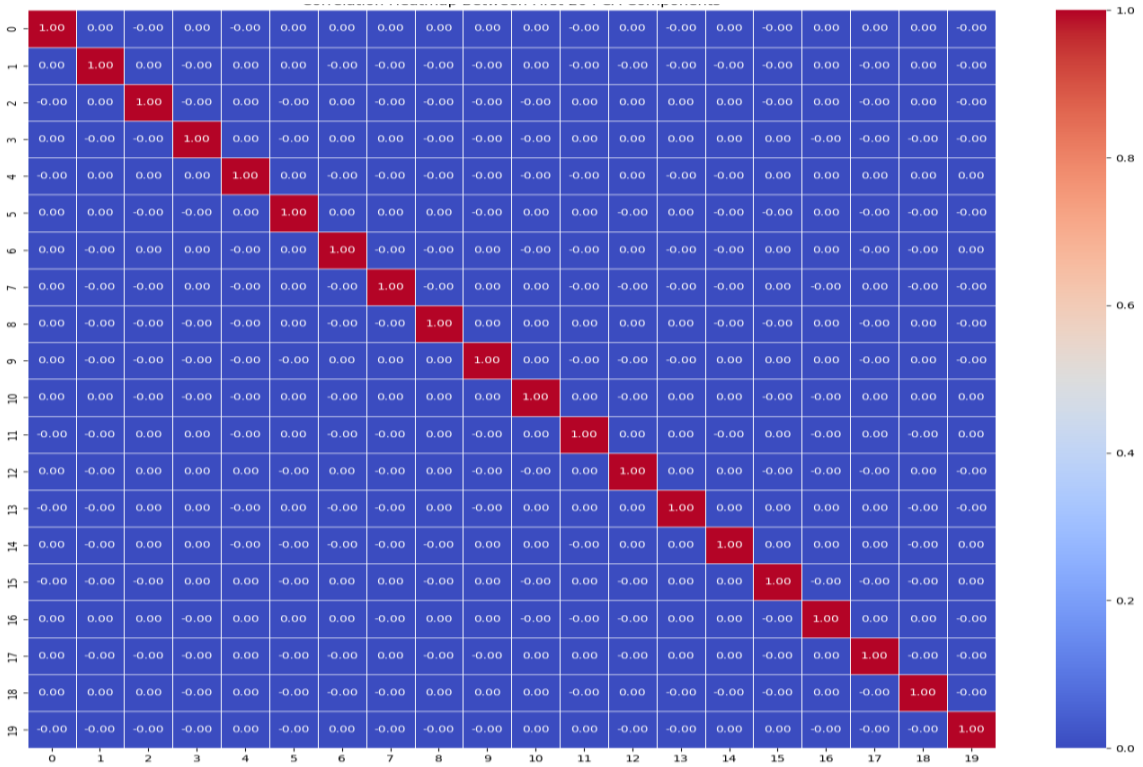


Fig. 6. Correlation between PCA Components (with self-correlation).

4) Train-test split: To evaluate model performance fairly, the dataset is divided into distinct subsets. For traditional machine learning models, the dataset is partitioned into 70% for training and 30% for testing to evaluate generalization performance on unseen data. In the case of deep learning models such as LSTM and DNN, the dataset is further divided into 70% for training and 30% for testing, enabling effective monitoring during training and ensuring robust model evaluation.

D. Modeling

1) Machine learning models: Random Forest (RF), KNN, XGBoost, and SVM are the most effective ML models for the detection of intrusions. The models are good for classification with feature-extracted input but are inefficient with complex temporal relationships within sequential input.

a) K-Nearest Neighbor (KNN): The KNN algorithm is a supervised algorithm in ML most commonly used to classify tasks [11]. It identifies the unlabeled data by considering the label and the available training data's features. The algorithm identifies the data by determining the point's nearest neighbors and the final label by majority voting. Among all the algorithms in ML, the algorithm in the case of the KNN algorithm is unique in terms of ease and interpretability and, in turn, acts to be a commonly used algorithm to classify tasks [12]. Even despite the ease, the algorithm performs exceptionally in the case of solving the classification and the regression problems in different datasets, regardless of size, label distribution, the datasets' noise, and the datasets' ranges [12].

b) Random Forests (RF): RF is a versatile algorithm in ML, famous for reducing the effect of overfitting, a prevalent problem in decision trees (DTs). It performs tasks such as classification, regression, and others by building many DTs during the training period. The algorithm works by evaluating multiple distinct decision trees (DTs) and making the prediction through a voting mechanism. Unlike in the case of a standard DT, in which each internal node gets partitioned by the optimal attribute, RF uses the optimal attribute from a random subset of predictors at each internal node. This randomization serves to enhance model generalization and resistance, and the resulting algorithm is a versatile tool in numerous applications in ML.

c) XGBoost (XGB): XGBoost (XGB) is a Gradient Tree Boosting algorithm powerful enough to solve heavy-scale ML problems in an efficient and effective manner. It possesses great prediction precision and model training speed, and it's the leading performer in all the competitions at Kaggle. The mechanism in XGB lies in the addition of trees in an iterative fashion and the division of the features during the course of the expansion in the tree. The model learns to fit the residuals from the last prediction each time a new tree gets added [13]. Given an input x_i, a true label y_i, and a raw prediction z_i before applying the sigmoid function, according to [14], the XGBoost model defines its objective function in the following equation:

$$L^{(t)} = \sum_{i=1}^{n} \quad l\left(y_i, Z_i^{(t-1)} + f_t(x_i)\right) + \Omega(f_t) + c \quad (1)$$

Where $l(.,.)$ represents the loss function, t denotes the t-th tree, and Ω serves as a penalty for model complexity. The term $\Omega(f_t)$ refers to the regularization penalty, while c is a constant.

The second-order Taylor expansion is given by:

$$f(x+\Delta x) \approx f(x) + f'(x)\Delta x + 1/2\, f''(x)\Delta x^2 \quad (2)$$

By substituting Eq. (2) into Eq. (1), we can obtain the following result.

$$L^{(t)} \approx \sum_{i=1}^{n}\left[l\left(y_i + Z_i^{(t-1)}\right) + g_i f_t(x_i) + \frac{1}{2}h_i\left(f_t(x_i)\right)^2\right] \quad (3)$$

where $g_i = \partial\, L/\partial\, z_i$, and $h_i = \partial^2\, L/\partial\, z_i^2$. By eliminating the constant terms, we derive the following simplified objective at step t.

$$L^{(t)} \approx \sum_{i=1}^{n}\left[g_i f_t(x_i) + \frac{1}{2}h_i\left(f_i(x_i)\right)^2\right] + \Omega(f_t) \quad (4)$$

The terms g_i and h_i play a vital role in optimizing the XGBoost training process. For binary classification, the model typically employs cross-entropy (CE) as its default loss function.

$$L = -\sum_{i=1}^{n}[y_i log(\acute{y}_i) - (1-y_i)log(1-\acute{y}_i)] \quad (5)$$

In Eq. (5), $\acute{y}_i = 1/[1+exp(-z_i)]$, that is sigmoid is selected as activation. Therefore, we can get:

$$\partial\, \acute{y}_i/\partial\, z_i = \acute{y}_i(1-\acute{y}_i) \quad (6)$$

2) Deep learning models: Deep-learning methods often fall under the umbrella of unsupervised pre-trained networks—architectures that stack many more layers and parameters than typical machine-learning neural nets, earning them the designation "deep."

a) Deep Neural Network (DNN): An Artificial Neural Network (ANN) is a model based on the structure and function of the brain [15]. Since neural networks (NN) are powerful nonlinear discriminators in the event of problems in classification, because they are able to describe any decision boundary in the feature space [16]. In recent years, Deep Neural Networks (DNNs) gained significant interest in intrusion detection research and evolved from Shallow Neural Networks (SNNs). The feature abstraction ability in DNNs and the ability to represent highly complex patterns make them extremely useful in applications in DL. Because of their ability to represent data in a good way, DNNs are in high demand in order to design efficient and robust solutions.

The results are produced in a DNN based on the connection weights and activation functions in the neurons. The DNN is composed of multiple processing layers, and every layer contributes to decision-making and feature extraction. Several hyperparameters dictate the operation of a DNN and are to be determined in advance, including the number of units, number of layers, weights and bias initializers, activation function, regularizer's coefficient, learning rate, and the optimizer. In this DNN model, ReLU activation is applied in the input layer and in every hidden layer. The ReLU function is a piecewise linear function and returns the same input in the situation where the input is a positive number and a value of zero in the situation where the input is a negative number [17]. The neurons activated by this function are also rectified linear activation units.

$$ReLU(x) = max(0, x) \quad (7)$$

b) Long Short-Term Memory (LSTM): The LSTM layer is a special kind of [18] RNN, and the main job of the LSTM layer is to handle sequence data with temporal relationships, such as text and relation data. The LSTM layer consists of three gates, the forget gate, the output gate, and the input gate, and the shared state. The use equation and the LSTM layer's detailed working are presented in the following equations:

$$f_s = \sigma\left(W_f \cdot [h_{s-1}, x_s] + b_f\right) \quad (8)$$

$$i_s = \sigma\left(W_f \cdot [h_{s-1}, x_s] + b_i\right) \quad (9)$$

$$\acute{C} = tanh(W_c \cdot [h_{s-1}, x_s] + b_C) \quad (10)$$

$$C_s = f_s \cdot C_{s-1} + i_s \cdot \acute{C} \quad (11)$$

$$o_s = (W_o[h_{s-1}, x_s] + b_C) \quad (12)$$

$$h_s = o_s \cdot tan(C_s) \quad (13)$$

The input gate controls how the LSTM cell state acquires the information, in Eq. (9). The forget gate controls how the LSTM cell state forget the information, in the Eq. (8). The cell state updates by the Eq. (10) and Eq. (11).

Output Layer:

The output layer generates the predicted sentiment classification of the comment. The final output vector is scaled using the softmax activation function to produce a probability distribution across each class. The equation is as follows:

$$y = softmax\left(W h_{fc} + b\right) \quad (14)$$

E. Evaluation Metrics

1) Confusion matrix: It's a matrix to represent how model classify dataset, also to use to check how good the model's performance of classification. It's comprised by four components.

In a confusion-matrix context, true positives (TP) are cases where an instance is genuinely positive and the model correctly labels it as such, whereas true negatives (TN) are instances that are truly negative and rightly classified as negative. By contrast, false positives (FP) occur when a genuinely negative instance is mistakenly flagged as positive, and false negatives (FN) arise when a genuinely positive instance is incorrectly marked as negative.

2) ROC Curve (Receiver Operating Characteristic Curve): A graph that shows classification performance across all thresholds by plotting the true-positive rate (TPR) against the false-positive rate (FPR), illustrating the trade-off between sensitivity and specificity.

3) Score: The harmonic mean of precision and recall; a single, balanced metric that is especially informative when class distributions are unbalanced. It ranges between 0 and 1, and 1 in the event of optimal precision and recall.

$$\text{F1 Score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (15)$$

4) Accuracy: The rate of correctly classified observations over the total observations. It is a helpful indicator in the situation where false positives and false negatives are equally distributed.

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} \quad (16)$$

5) Precision: The proportion of the total predicted positives actually being positives. The higher the precision, the lower the rate of the incorrect positives, and the lowest the incorrect positives.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \quad (17)$$

6) Recall: The ratio of correctly predicted positive instances to all actual positive instances. A higher recall indicates fewer missed positives and a lower rate of false negatives.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \quad (18)$$

IV. Results

This section presents how a range of ML and DL models performed on binary and multi-class intrusion-detection tasks using the "NF-UNSW-NB15" dataset. Before training, we applied PCA to cut dimensionality and boost both speed and accuracy. Every model was tuned with the same hyper-parameter settings to keep the comparison fair. Their effectiveness was gauged with a full suite of metrics—accuracy, precision, recall, F1-score, AUC, and confusion matrices—so we could see each algorithm's strengths and weaknesses in detail. Results are split into two categories: binary detection of normal versus malicious traffic, and multi-class detection that pinpoints the exact attack type.

A. Hyperparameter Configuration

All deep learning models (LSTM and DNN) were trained using a fixed set of hyperparameters to ensure a fair and consistent comparison. For binary classification tasks, the loss function employed was binary crossentropy, while sparse categorical crossentropy was used for multi-class classification. Model optimization was performed using the Adam optimizer with a learning rate of 0.001. Both tasks were trained over 20 epochs with a batch size of 128. Additionally, early stopping was applied based on validation loss to prevent overfitting and ensure optimal generalization performance.

Table IV presents the hyperparameter settings used for model training, including learning configuration, optimization strategy, and loss functions for both binary and multi-class tasks.

B. Binary Classification Results

Before applying SMOTE, all models achieved high overall accuracy (0.99); however, recall and F1-scores—particularly for the minority class—were comparatively lower. For instance, the LSTM model achieved a recall of 0.92 and an F1-score of 0.94, while the DNN model recorded a recall of 0.90. These results indicate reduced sensitivity to minority class detection due to the dataset's class imbalance.

Moreover, AUC scores, which evaluate a model's ability to distinguish between classes, were also affected. Deep learning models such as DNN and LSTM yielded relatively lower AUCs in the range of 0.90–0.92, suggesting less reliable separability between benign and malicious traffic. In contrast, ensemble models like Random Forest and XGBoost performed better, achieving AUC scores around 0.98. Nevertheless, even these models demonstrated measurable improvements after applying SMOTE.

These findings confirm that SMOTE plays a critical role not only in enhancing recall and F1-scores but also in improving the overall discriminative power of classifiers, as reflected in AUC metrics. Table V summarizes the performance of LSTM, DNN, Random Forest, KNN, and XGBoost on the binary classification task prior to applying SMOTE.

After applying SMOTE, all models demonstrate strong performance, with Random Forest and KNN achieving high accuracy and F1-scores. While LSTM and DNN yield slightly lower precision for the Attack class, they still achieve perfect recall, indicating high sensitivity to positive cases. The AUC score of 0.99 or higher across all models confirms excellent separability between classes. Among all models, RF achieved the best overall performance with an F1-score of 0.97, and highly balanced precision and recall values, especially for class 1 (Attack class). Its confusion matrix is very high, showing low misclassificationwith 2308 false positives and 199 false negative, Fig. 7 shows Random Forest confusion matrix that effectively classified both Normal and Attack classes, making it the most reliable model for the binary classification task.

TABLE IV. Hyperparameter Settings

Parameter	Value
Epochs	20
Batch Size	128
Learning Rate	0.001
Optimizer	Adam
Loss Function (Binary)	Binary Crossentropy
Loss Function (Multi)	Sparse Categorical Crossentropy
PCA Components (Binary and Multi)	20

TABLE V. Binary Classification Performance without SMOTE

Model	Accuracy	Precision	Recall	F1-score	AUC
LSTM	0.99	0.97	0.92	0.94	0.92
DNN	0.99	0.97	0.90	0.93	0.90
RF	0.99	0.97	0.98	0.98	0.98
KNN	0.99	0.95	0.95	0.94	0.95
XGBoost	0.99	0.96	0.97	0.97	0.98

Fig. 8 shows LSTM confusion matrix for classifying both Normal and Attack classes.

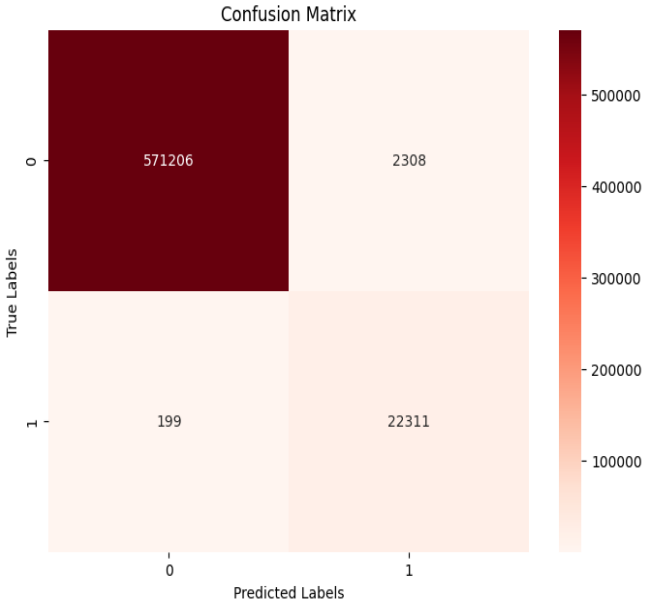


Fig. 7. RF confusion matrix.

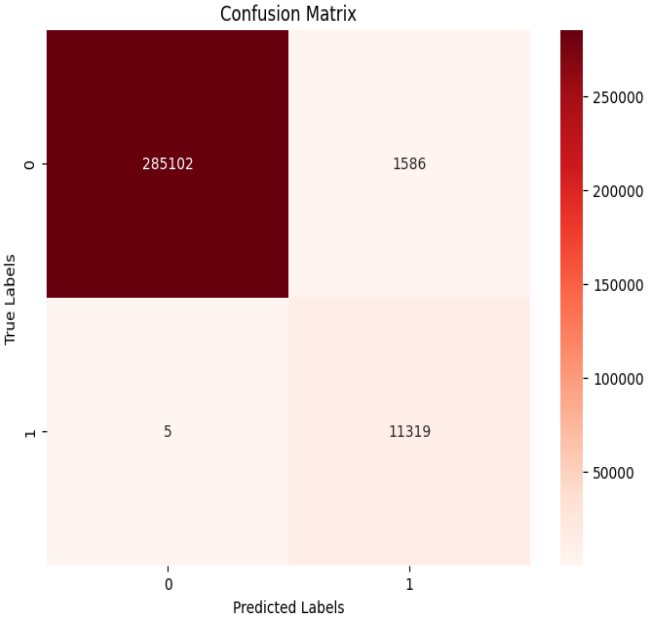


Fig. 8. LSTM confusion matrix.

The confusion matrix of the LSTM model reveals strong sensitivity in detecting attacks, although a slight increase in false positives indicates a trade-off in precision.

Fig. 9 shows DNN confusion matrix for classifying both Normal and Attack classes.

This matrix illustrates the DNN model's good performance, with solid detection capability and also slight false alarm rate, reflecting effective learning of attack patterns.

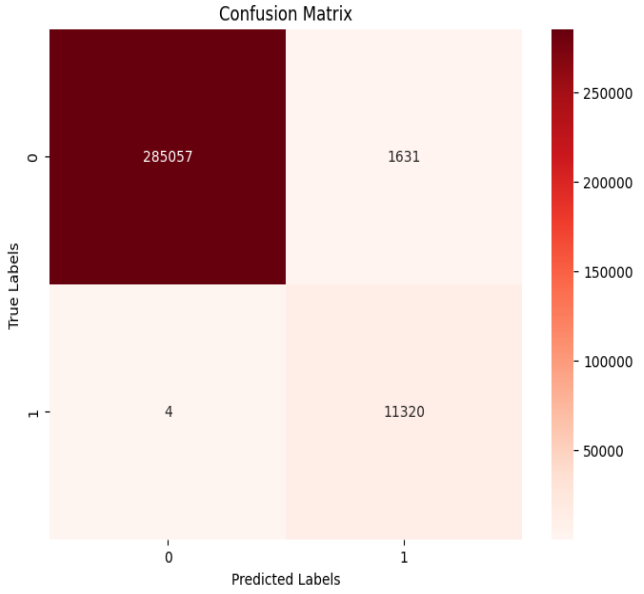


Fig. 9. DNN confusion matrix.

Fig. 10 shows KNN confusion matrix for classifying both Normal and Attack classes.

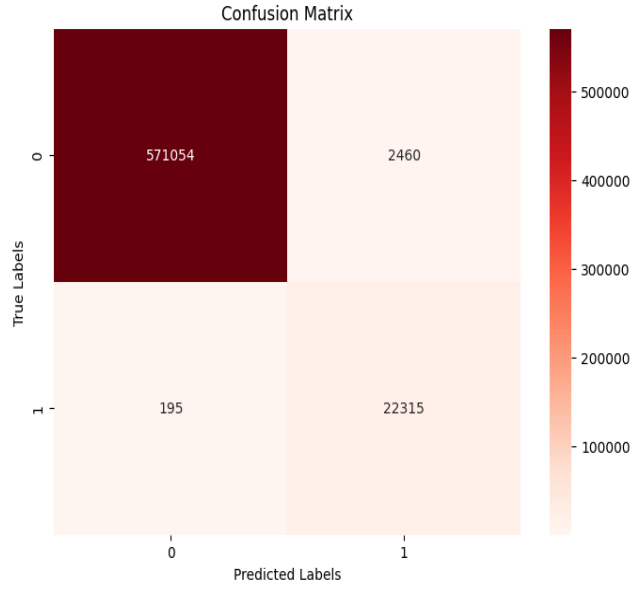


Fig. 10. KNN confusion matrix.

The KNN confusion matrix indicates good accuracy, with slightly higher misclassification of Normal status.

Fig. 11 shows XGBoost confusion matrix for classifying both Normal and Attack classes.

XGBoost exhibits high precision and recall, evident in its compact and clearly defined confusion matrix blocks with 2164 false positives and 239 false negatives, signifying high discriminative power.

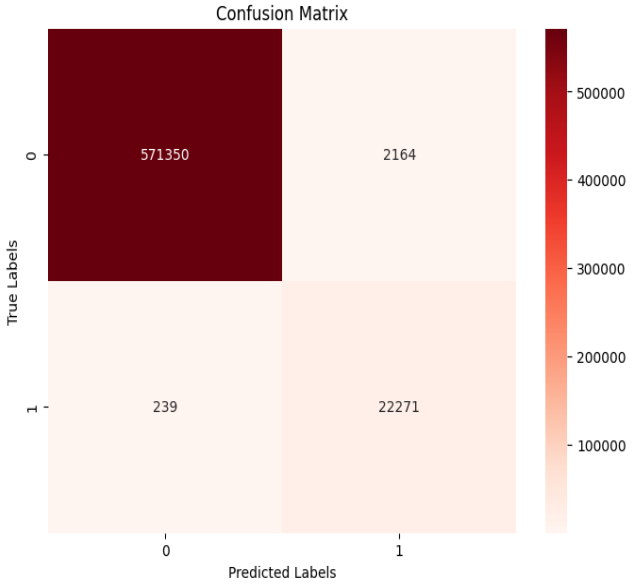


Fig. 11. XGBoost confusion matrix.

Table VI summarizes the performance of LSTM, DNN, RF, KNN, and XGBoost on the binary classification task.

TABLE VI. BINARY CLASSIFICATION PERFORMANCE WITH SMOTE

Model	Accuracy	Precision	Recall	F1-score	AUC
LSTM	0.99	0.94	0.99	0.97	0.99
DNN	0.99	0.94	0.99	0.96	0.99
RF	0.99	0.95	0.99	0.97	0.99
KNN	0.99	0.95	0.99	0.97	0.99
XGBoost	0.99	0.95	0.99	0.97	0.99

C. Multi-Class Classification Results

Unlike binary classification, the multi-class classification task suffered from noticeable performance degradation in the absence of SMOTE, especially in models like DNN, which recorded extremely low precision, recall, and F1-score (all = 0.10) despite an overall accuracy of 0.99. This illustrates that accuracy alone is misleading under class imbalance.

Similarly, LSTM showed moderate metrics (F1 = 0.52), while Random Forest and XGBoost performed relatively better (F1 = 0.64 and 0.62 respectively). AUC scores also reflected this trend—DNN had a poor AUC of 0.50, while XGBoost reached 0.98. These results demonstrate the limitations of models when trained on highly imbalanced data and reinforce the need for resampling methods like SMOTE to achieve fair multi-class performance across all attack types.

Table VII summarizes the performance of LSTM, DNN, RF, KNN, and XGBoost on the multi-class classification task before applying SMOTE.

Table VIII details the multi-class classification performance across the same set of models after SMOTE.

Model performances vary more noticeably in the multi-class setting. LSTM and DNN, while achieving high overall accuracy due to class imbalance in test set, show lower macro-averaged F1-scores, indicating challenges in learning minority classes. In contrast, Random Forest and XGBoost perform robustly across all classes, benefiting from ensemble-based learning. KNN also performs competitively but with slightly lower recall. The XGBoost classifier emerged as a competitive performer in the multi-class setting, achieving a macro F1-score (0.71), reflecting moderate performance across most classes, including minority ones. The confusion matrix of XGBoost reflects moderate performance on minority classes such as 6, 7, 8, and 9 with misclassifications, e.g., 919 false negatives for class 2 and 1455 for class 4 as shown in Fig. 12.

LSTM's multi-class confusion matrix (Fig. 13) shows robust classification for dominant classes but reveals some difficulty in differentiating among minority attack types.

TABLE VII. MULTI-CLASS CLASSIFICATION MACRO-AVERAGED SCORES WITHOUT SMOTE

Model	Accuracy	Precision	Recall	F1-score	AUC
LSTM	0.99	0.57	0.53	0.52	0.92
DNN	0.99	0.10	0.10	0.10	0.50
RF	0.99	0.66	0.62	0.64	0.94
KNN	0.99	0.61	0.58	0.59	0.87
XGBoost	0.99	0.66	0.60	0.62	0.98

TABLE VIII. MULTI-CLASS CLASSIFICATION MACRO-AVERAGED SCORES

Model	Accuracy	Precision	Recall	F1-score	AUC
LSTM	0.99	0.62	0.56	0.56	0.99
DNN	0.99	0.66	0.57	0.59	0.99
RF	0.98	0.68	0.71	0.63	0.99
KNN	0.99	0.69	0.66	0.63	0.89
XGBoost	0.99	0.73	0.80	0.71	0.99

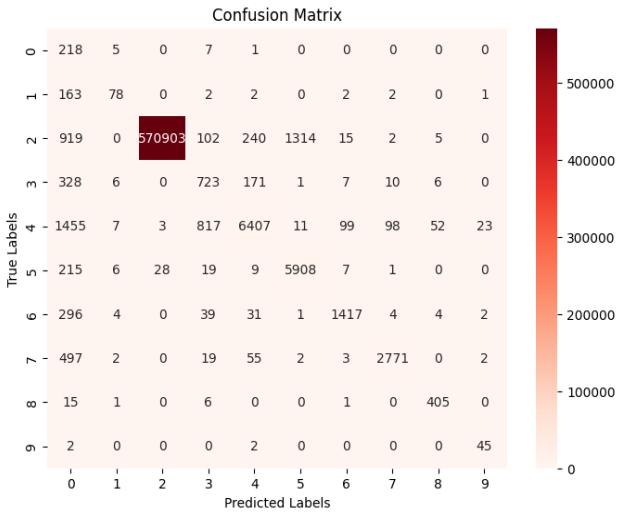


Fig. 12. XGboost confusion matrix in multi-classification.

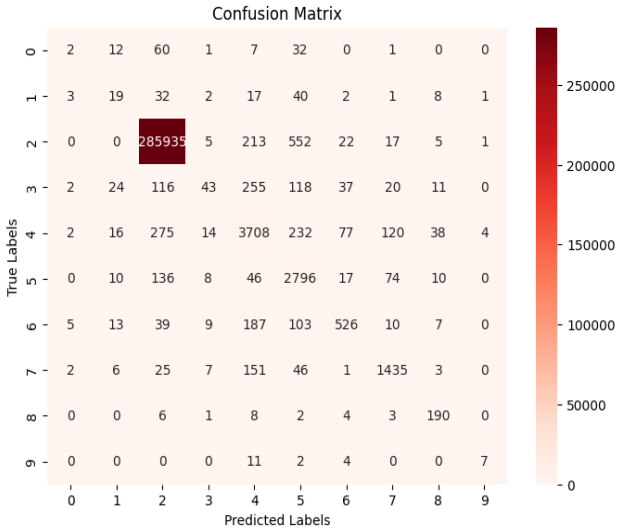


Fig. 13. LSTM confusion matrix in multi-classification

The DNN model performs well across several classes, but the confusion matrix reflects misclassification among minority classes as shown in Fig. 14.

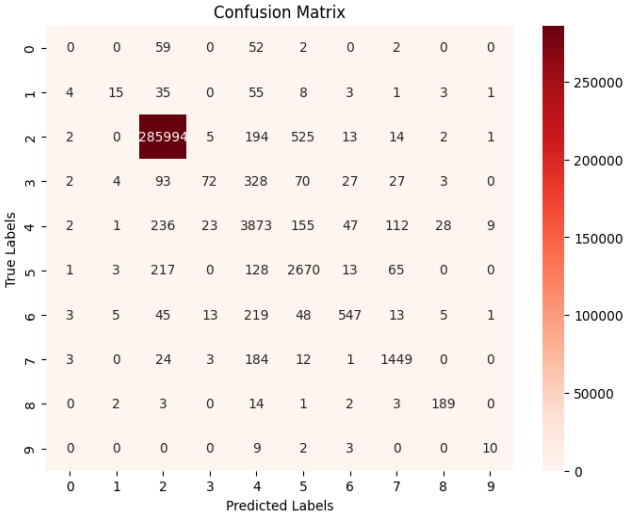


Fig. 14. DNN confusion matrix in multi-classification.

The confusion matrix of KNN (Fig. 15) shows good effectiveness, better than LSTM and DNN, but still less than Xgboost.

Random Forest delivers a compact and efficient confusion matrix, excelling at distinguishing between multiple attack types, especially frequent ones, while showing some overlap in less-represented classes as shown in Fig. 16.

D. Inference Time Evaluation for Real-Time Applicability

In addition to classification performance, prediction time is a critical consideration for real-world deployment, particularly in systems requiring real-time or near-real-time responses. Table IX presents both the total and average prediction times (per 1,000 records) for each model across binary and multi-class classification tasks, providing insight into their computational efficiency.

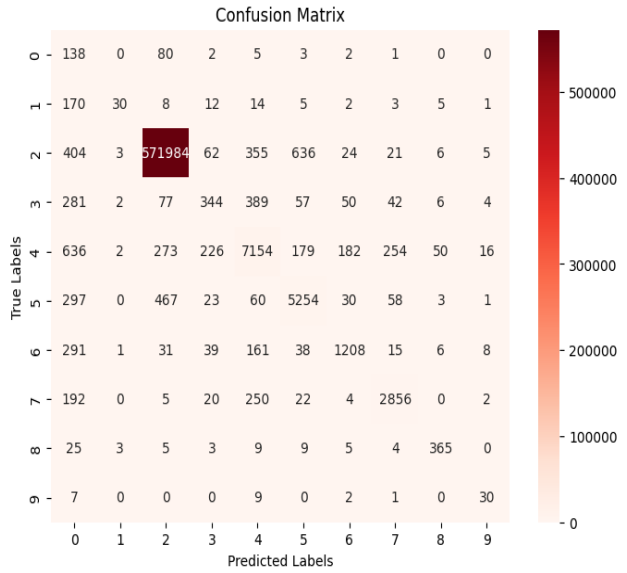


Fig. 15. KNN confusion matrix in multi-classification.

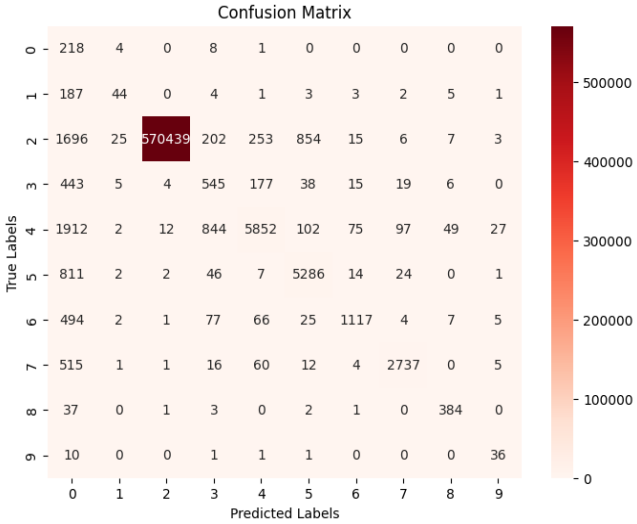


Fig. 16. Random forest confusion matrix in multi-classification.

TABLE IX. PREDICTION TIME FOR EACH CLASSIFIER

Model	Task	Total Time (s)	Avg. Time per 1000 Records (s)
LSTM	Binary	25	0.3575
DNN	Binary	20	0.0700
RF	Binary	4.519	0.0076
KNN	Binary	73.4142	0.1232
XGBoost	Binary	14.323	0.0240
LSTM	Multi-class	22.8	0.07
DNN	Multi-class	20.5489	0.0690
RF	Multi-class	5.0694	0.0128
KNN	Multi-class	39.0577	0.0983
XGBoost	Multi-class	26.7887	0.0449

Random Forest exhibited the fastest prediction times in both binary and multi-class tasks, making it highly suitable for real-time applications. Random Forest also demonstrated strong efficiency, whereas deep learning models, particularly LSTM and KNN, had the longest processing times.

V. Discussion

This study presents a comprehensive evaluation of multiple machine learning (ML) and deep learning (DL) classifiers for both binary and multi-class network intrusion detection using the NF-UNSW-NB15 dataset. The results demonstrate that the proposed models—particularly XGBoost and Random Forest—achieved superior performance, with macro-averaged F1-scores and accuracy exceeding 99% in binary classification. These models also maintained robustness in multi-class scenarios, achieving macro F1-scores of up to 0.71.

In contrast, the recent work by Samantaray et al. [9], which employed traditional ML algorithms on the same dataset, reported significantly lower macro F1-scores—only 0.26 with Logistic Regression and 0.17 with Random Forest—despite using MaxAbsScaler for feature scaling. Although their approach achieved relatively high overall accuracy (up to 94%) in multi-class classification, the low macro-averaged metrics reflect poor generalization across minority classes, indicating that class imbalance was not sufficiently addressed.

TABLE X. Comparative Results of Best Classifiers Across Studies

Study	Model	Dataset	Type	Accuracy	Macro Precision	Macro Recall	Macro F1-Score
This Work	RF	NF-UNSW-NB15-v2	Binary	0.99	0.95	0.99	0.97
This Work	XGBoost	NF-UNSW-NB15-v2	Multi-class	0.99	0.73	0.80	0.71
Kasongo and Sun [4]	XGBoost	NF-UNSW-NB15	Binary	0.91	0.80	0.98	0.88
Samantaray et al. [9]	RF	NF-UNSW-NB15	Multi-class	0.90	0.23	0.20	0.17
Samantaray et al. [9]	KNN	NF-UNSW-NB15	Multi-class	0.93	0.29	0.31	0.26
Sayed et al. [10]	IoTCNN	NF-UNSW-NB15-v2	Multi-class	~0.99	~0.42	~0.63	~0.44
Sharma et al. [33]	RF	UNSW-NB15	Binary	0.87	0.90	0.86	0.86

Similarly, the CNN-based models proposed by Sayed et al. [10], namely IoTCNN and MyCNN, demonstrated limited macro-level performance. For example, the IoTCNN model produced low precision and F1-scores in most attack categories, including F1 = 0.14 for Analysis and 0.13 for DoS, despite exhibiting high overall accuracy. These results suggest that while convolutional models may effectively capture local patterns, they often struggle with inter-class variance and generalization without advanced preprocessing or data augmentation strategies.

In addition to Samantaray et al. [9] and Sayed et al. [10], we include the study by Sharma et al. [33], which evaluates ensemble methods on UNSW-NB15 using a binary setting. Their best model (Random Forest) achieves Accuracy = 0.87, Macro Precision = 0.90, Macro Recall = 0.86, and Macro F1 = 0.86, whereas our binary IDS on NF-UNSW-NB15-v2 attains Accuracy ≈ 0.99 and Macro F1 ≈ 0.97. Thus, our approach improves binary macro-level performance by ≈ 11 points in F1 while also raising accuracy by ≈ 12 percentage points, reinforcing that the proposed SMOTE + PCA + diverse ML/DL pipeline generalizes better under class imbalance than strong ensemble baselines.

Overall, our findings highlight the effectiveness of ensemble models—such as XGBoost and Random Forest—in capturing complex feature relationships and addressing class imbalance. A comparative summary of results is provided in Table X.

VI. Conclusion

This work developed and empirically validated an end-to-end intrusion-detection pipeline that couples rigorous preprocessing (StandardScaler normalization, SMOTE oversampling, and PCA dimensionality reduction) with five classifiers (XGBoost, RF, KNN, DNN, and LSTM) evaluated on the "NF-UNSW-NB15-v2" dataset. The experiments covered both binary (normal vs attack) and nine-class attack identification tasks, using a unified hyper-parameter budget to ensure fair comparison.

Results underscore the strength of ensemble trees for tabular network-flow data: XGBoost and Random Forest achieved high metrics (≈ 0.9960 for accuracy, precision, recall, F1, and AUC) in binary detection, while XGBoost retained a competitive macro F1-score (0.71) and a balanced confusion matrix in multi-class testing. Deep models matched overall accuracy but lagged in macro scores, indicating residual sensitivity to minority classes. Crucially,RF and XGBoost's inference latency—on the order of milliseconds per thousand flows—demonstrates that top-tier accuracy can coexist with real-time throughput, making the approach deployable in production networks.

Although the pipeline raises the state of the art for this dataset, two limitations remain: extremely rare classes such as Worms and Analysis still risk misclassification despite SMOTE, and features were restricted to NetFlow attributes. Future work should explore cost-sensitive or generative resampling, integrate payload-level and temporal-correlation features, and test hybrid architectures (e.g., LSTM embeddings feeding XGBoost) under online-learning and edge-deployment constraints. Overall, the study provides a reproducible blueprint and a strong baseline for practitioners seeking accurate, low-latency, and resource-efficient intrusion detection in modern networked environments.

Acknowledgment

The authors extend their appreciation to the Deanship of Research and Graduate Studies at King Khalid University for funding this work through Large Research Project under grant number RGP2/318/46.

REFERENCES

[1] L. Cadieux, "20 Shocking Cybercrime Statistics: 2024 Edition," The Devolutions Blog, 4 July 2024. [Online]. Available: https://blog.devolutions.net/2024/07/20-shocking-cybercrime-statistics-2024-edition/?utm_source=chatgpt.com. [Accessed 1 Mars 2025].

[2] M. Sweney, "BT identifying 2,000 signals a second indicating possible cyber-attacks," The Guardian, 12 September 2024. [Online]. Available: https://www.theguardian.com/business/2024/sep/12/hackers-weaponising-ai-for-cybercrime-bt-warns?utm_source=chatgpt.com. [Accessed 1 Mars 2025].

[3] O. Ogundairo and P. Broklyn, "Machine Learning Algorithms for Intrusion Detection Systems," Journal of Cyber Security, 2024.

[4] S. Kasongo and Y. Sun, "Performance analysis of intrusion detection systems using a feature selection method on the UNSW-NB15 dataset," Journal of Big Data, vol. 7, no. 1, p. 105, 2020.

[5] V. Kumar, D. Sinha, A. Das, D. S. Pandey and R. Goswami, "An integrated rule based intrusion detection system: analysis on UNSW-NB15 data set and the real time online dataset," Cluster Computing, vol. 23, 2020.

[6] S. More and R. Kalkundri, "Evaluation of deceptive mails using filtering & WEKA," in 2015 International Conference on Innovations in Information,Embedded and Communication Systems (ICIIECS), 2015.

[7] R. Tahri, Y. Balouki, A. Jarrar and L. Abdellatif, "Intrusion Detection System Using machine learning Algorithms," ITM Web of Conferences, vol. 46, 2022.

[8] U. S. Musa, M. Chhabra, A. Ali and M. Kaur, "Intrusion Detection System using Machine Learning Techniques: A Review," in Int. Conf. Smart Electron. Commun. (ICOSEC 2020), 2020.

[9] M. Samantaray, R. C. Barik and A. K. Biswal, "A comparative assessment of machine learning algorithms in the IoT-based network intrusion detection systems," Decision Analytics Journal, vol. 11, p. 100478, 2024.

[10] N. Sayed, M. Shoaib, W. Ahmed, S. N. Qasem, A. M. Albarrak and F. Saeed, "Augmenting IoT Intrusion Detection System Performance Using Deep Neural Network," Computers, Materials & Continua, vol. 72, no. 2, p. 3511–3534, 2022.

[11] D. Bzdok, M. Krzywinski and N. Altman, "Machine learning: supervised methods," Nat. Methods, vol. 15, p. 5–6, 2018.

[12] S. Uddin, I. Haque and H. Lu, "Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction," Sci Rep, vol. 12, no. 6256, 2022.

[13] S. He, B. Li, H. Peng, J. Xin and E. Zhang, "An Effective Cost-Sensitive XGBoost Method for Malicious URLs Detection in Imbalanced Dataset," IEEE Access, vol. 9, pp. 93089-93096, 2021.

[14] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in 22nd ACM SIGKDD International Conference, 2016.

[15] S. Jukic, M. Saracevic, A. Subasi and J. Kevric, "Comparison of ensemble machine learning methods for automated classification of focal and non-focal epileptic EEG signals," Mathematics, vol. 8, no. 9, p. 1481, 2020.

[16] S. Zhang, Z. Shen, and H. Yang, "Deep Network Approximation: Achieving Arbitrary Accuracy with Fixed Number of Neurons," Journal of Machine Learning Research, vol. 23, no. 276, pp. 1–60, 2022.

[17] J. Brownlee, "A Gentle Introduction to the Rectified Linear Unit (ReLU)," Machine Learning Mastery, 20 August 2020. [Online]. Available: https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/. [Accessed 1 March 2025].

[18] F. Huang, X. Li, C. Yuan, S. Zhang, J. Zhang and S. Qiao, "Attention-Emotion-Enhanced Convolutional LSTM for Sentiment Analysis," IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 9, pp. 1-14, 2021.

[19] "Machine Learning-Based NIDS Datasets," The University of Queensland, [Online]. Available: https://staff.itee.uq.edu.au/marius/NIDS_datasets/#RA1. [Accessed 1 March 2025].

[20] M. Sarhan, S. Layeghy and M. Portmann, "Towards a Standard Feature Set for Network Intrusion Detection System Datasets," arXiv, 2021.

[21] Nour Moustafa and Jill Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems," Military Communications and Information Systems Conference (MilCIS), 2015.

[22] M. Hopkins, E. Reeber, G. Forman, and J. Suermondt, Spambase Dataset, UCI Machine Learning Repository, 1999.

[23] I. Androutsopoulos et al., "An experimental comparison of naive Bayesian and keyword-based anti-spam filtering with personal e-mail messages," SIGIR Conference, 2000.

[24] B. Klimt and Y. Yang, "The Enron Corpus: A New Dataset for Email Classification Research," European Conference on Machine Learning (ECML), 2004.

[25] I. Androutsopoulos et al., PU1 and PU2 corpora for spam filtering experiments, 2000.

[26] M. Tavallaee, E. Bagheri, W. Lu, and A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," IEEE Symposium on CISDA, 2009.

[27] KDD Cup 1999 Dataset. UCI Machine Learning Repository.

[28] J. Song and H. Kim, "Network intrusion detection based on semi-supervised learning and clustering," Expert Systems with Applications, 2013.

[29] C. Kolias, G. Kambourakis, A. Stavrou, and S. Gritzalis, "Intrusion detection in 802.11 networks," IEEE ICC, 2015.

[30] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," ICISSP, 2018.

[31] M. Fernandez, J. Garcia, A. Sanz, and J. E. Diaz-Verdejo, "UGR'16: A new dataset for evaluation of IDSs," Computer Networks, 2018.

[32] D., Elreedy, A. F., Atiya, & F., Kamalov, "A theoretical distribution analysis of synthetic minority oversampling technique (SMOTE) for imbalanced learning," *Machine Learning*, 113:4903–4923, 2024.

[33] N. Sharma, N. S. Yadav, and S. Sharma, "Classification of UNSW-NB15 dataset using Exploratory Data Analysis using Ensemble Learning," EAI Endorsed Transactions on Industrial Networks and Intelligent Systems, vol. 8, no. 29, e4, Oct. 2021.

Comprehensive Analysis of Machine and Deep Learning Models for Stock Market Prediction

Hadi S. AlQahtani, Prof. Mohammed J. Alhaddad, Dr. Mutasem Jarrah
Department of Information Technology-Faculty of Computing and Information Technology,
King Abdulaziz University, Jeddah, Saudi Arabia

***Abstract*—Stock market prediction is a core task in financial engineering that requires sophisticated methods to extract subtle market and volatility trends. The increasing complexity of the stock market has led to the integration of advanced machine learning (ML) and deep learning (DL) techniques to improve accuracy beyond traditional statistical methods. This research provides a taxonomy of stock market prediction methods and reviews key regression-based models, including linear regression and advanced neural networks like recurrent neural networks (RNNs), long short-term memory (LSTM), and hybrid (CNN-LSTM) models. The study deploys and evaluates three specific models: Linear Regression, RNNs, and LSTMs. The models were trained and tested using modern data preprocessing procedures, including Z-score normalization and temporal sequencing. The findings show that the Linear Regression (LR) model performed better, with a Root Mean Square Error (RMSE) of 0.334 during training and 0.304 during testing, and a Mean Absolute Error (MAE) of 0.203 and 0.207, respectively. This contrasted with the deep learning models, which had higher error rates. The LSTM achieved a training RMSE of 0.355, while the RNN model had a training RMSE of 0.383. These results provide empirical evidence that increased model complexity does not necessarily translate into better forecasting accuracy in financial applications, and that model selection is both context-sensitive and data-driven. The findings mentioned the challenge of nonstationarity in stock market data and the need to periodically retrain models on recent data.**

Keywords—Deep learning; machine learning; prediction methods; stock market; regression; taxonomy

I. Introduction

Predicting stock prices accurately is crucial for investors, financial markets, and economy decision makers because of its profound implications for investment decisions. Two traditional approaches are used for forecasting stock trends, fundamental analysis and technical analysis. Fundamental analysis consists of the assessment of the actual value of securities based on the company's financial reports and other macroeconomic factors [1]. This method is more suitable for strategic long-term investment. Technical analysis, on the other hand, involves the use of price movements, volumes, and charts to determine the prognosis of future prices. This method is applied for short-term trading and help determine tendencies and the right time for trading operations [2].

The financial forecasting industry, particularly stock market prediction, has been greatly impacted by AI technologies as it continues to evolve rapidly. Traditional financial time series and time series analysis techniques often fail to analyze the market's complex nonlinear data, which made researchers shift their focus to more sophisticated AI technologies [3]. These include Recurrent Neural Networks (RNNs), Long Short-Term Memory networks (LSTMs), and even hybrid models such as LSTM-CNNs [4]. RNNs provide the most basic method to model temporal dependencies in sequential data, while LSTMs improve upon RNNs by introducing memory cell architectures to mitigate the vanishing gradient problem [3]. More recently, LSTM-CNN models have been developed which combine LSTMs with Convolutional Neural Networks (CNNs), leveraging both temporal and spatial feature extraction capabilities. This research focuses on the stock price forecasting AI models and aims to analyze their performance comparatively and identify the most optimal model for practical usage [4].

The traditional forecasting methods struggle to capture the complexities and uncertainties inherent in stock markets. However, AI technologies are modern methods that offer promising ways for enhancing prediction accuracy [13]. The algorithms offer the ability to automatically detect patterns, adapt to changing market conditions, process large volumes of data, including historical prices, technical indicators, financial news, social media sentiment, leveraging large datasets and detecting intricate patterns, especially machine learning (ML) and deep learning (DL) techniques [5]. By employing ML and DL algorithms such as linear regression, logistic regression, RNNs, LSTM, and LSTM-CNN, researchers have attempted to exploit the vast amount of market data to uncover hidden relationships and forecast future price movements [6]. This paper mentions stock market prediction methods taxonomy, with recent advancements in stock market prediction achieved through possible methods that can assist in stock market prediction. Generally, this paper provides insights into the effectiveness and reliability of the most important approaches in stock market prediction, informing stakeholders about their potential benefits and challenges in real-world and prediction applications.

The reminder sections of this paper are: proposed previous work in Section III as a literature review. Section IV discussed methodology, results and findings are evaluated in Section V, Critical analysis is given in Section VI and Section VII is the conclusion.

II. Stock Market Prediction Methods Taxonomy

Stock market prediction methods are divided into three categories: fundamental analysis, technical analysis, and algorithmic approaches.

A. Fundamental Analysis

Fundamental analysis relies on economic indicators, company performance metrics, and financial reports to assess the intrinsic value of a stock [1].

B. Technical Analysis

Technical analysis focuses on historical price patterns, trading volumes, and chart indicators to forecast future price movements [12].

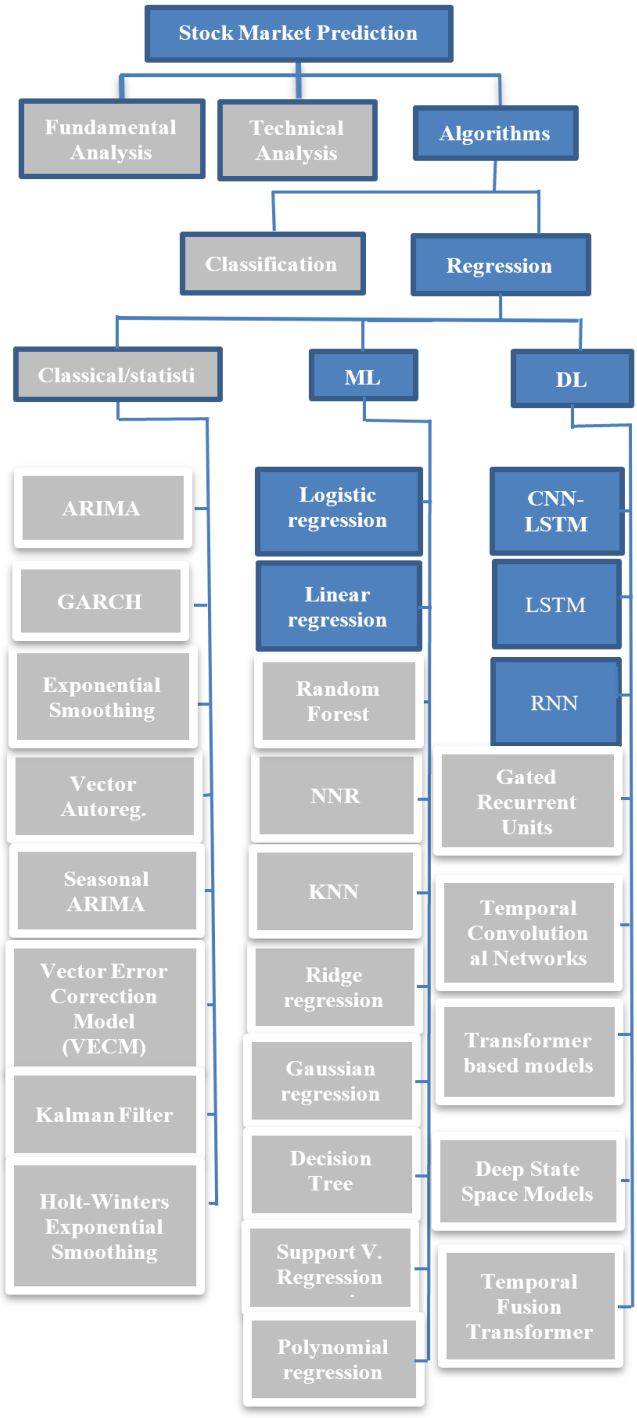


Fig. 1. Taxonomy of stock market prediction methods.

C. Algorithmic Approaches

Algorithmic approaches employ computational models and artificial intelligence techniques to identify patterns and generate predictions. Within algorithmic approaches, methods are typically organized into two main tasks: classification (e.g., predicting whether a stock will rise or fall) and regression (e.g., forecasting the exact stock price or return value) [6].

Since the present study is concerned with continuous stock price forecasting [15], the focus is placed on regression methods. These can be further classified into three different categories [14]:

- Classical/Statistical models (e.g., ARIMA, GARCH), which provide simple yet interpretable baselines.
- Machine Learning (ML) models (e.g., Linear Regression, Logistic regression, Support Vector Regression, Random Forests), which capture nonlinear relationships and improve predictive performance.
- Deep Learning (DL) models (e.g., RNN, LSTM, CNN-LSTM), which are designed to learn complex temporal dependencies and deliver state-of-the-art results for sequential financial data.

This hierarchical taxonomy provides a structured view of stock prediction methods, showing how approaches range from traditional econometric techniques to advanced AI-driven models. It also clarifies the scope of this study, which emphasizes regression-based methods across statistical, ML, and DL Models. These three categories contain more techniques, as shown in Fig. 1.

III. Literature Review

Predicting the stock market has attracted much interest from the scientific community, financial experts, and computer scientists. Future market prices and movements are likely to show complex nonlinear relationships with time that differ. Due to their availability and increased computational power, most technological tools are incredibly precise and elaborate.

The reviewed studies in this paper are organized into a taxonomy that highlights how stock market prediction methods are structured along the key dimensions: algorithm type

The discussed literature is limited to important machine learning (ML) approaches and deep learning (DL) approaches:

A. Machine Learning

1) Linear regression: Linear regression is treated as a supervised learning algorithm for regression tasks. It is an elementary method of predictive modeling, a subset of artificial intelligence, used to establish a relationship between an outcome variable and one or several predictor variables. It assumes that the best fit is a straight line that best predicts the errors between the predicted and actual values. The conditional assumptions are linearity, the absence of multicollinearity, and the normality and homoscedasticity of residuals. Research in [7] discussed regression methods used for prediction models in machine learning. The researchers overviewed definitions of the models, metrics for model

evaluation, assumptions, and methods of attribute selection. Some of the issues that were raised in the study include issues related to overfitting and the fact that while regression is a very powerful tool, it is, however, quite restricted in its ability to solve very complicated problems. Reference [8] designed a website that employed linear regression for stock price prediction. The researchers concluded that the performance on two years of historical prices provided reasonable accuracy. Authors of study [16] also noted that it is crucial to consider nonstationarity when dealing with stock data for production use. The model must be periodically retrained using more recent data to enhance its predictive ability. Research in [9] discussed how the linear regression model is trained for stock prediction using Python. The researchers explored exploratory data analysis, formulating a model, evaluating it, and incorporating it into trading strategies. Generally, they proved the existence of its kind but pointed out the dangers of oversimplification and the use of predictive modeling. Moreover, regression coefficients introduce interpretability measures to identify useful indicators [9]. Nonstationarity is one of the challenges faced by models and may be addressed by retraining on more recent sliding stock data. In general, literature supports the application of regression for stock prediction but simultaneously raises concerns about using it without proper precaution [8]. Important features are cross-validation, integration with other techniques, repetitiveness with retraining, interpretability and integration with human decision-making [14]. The greatest value is achieved when prescriptive analytics include predictions and insights into the trading systems.

2) Logistic regression: In machine learning and for classification tasks, logistic regression is a supervised learning algorithm. It is a specific feature of neural networks is that they require many training data points for accurate predictions. Neural networks can be used when making predictions based on binary outcomes. Logistic regression is one of the most frequently used in machine learning and can be utilized to predict a binary outcome point given predictor variable [14]. The study in [10] evaluated the abilities of logistic regression as a machine learning algorithms, including random forest and neural networks. This study revealed that logistic regression was not inferior to other algorithms after the method of choosing the number of features through the area under the ROC curve was applied [10]. This is where logistic regression proves to be a helpful interpretable baseline model. The study in [11] employed logistic regression and logitboost algorithms to build churn prediction models. Logit had a slightly better performance than logistic regression, although this came at the cost of performing hyperparameter tuning. Logistic regression provides an excellent and less complex solution to the model. Authors of [17] considered the rationale for employing logistic regression or linear regression analysis when analyzing treatment effects in experimental research with binary data. The study in [11] concluded that logistic regression was reasonable for disentangling treatment impact from risk profiles relevant to the outcome, even in situations of imbalanced randomization. Generally, the reviewed research proves the suitability of the logistic models for causal conclusions. Logistic regression has several benefits, namely, interpretability, efficiency, and reliability of the results. Coefficients of logit show how each predictor is associated with the outcome variable [11]. Fewer tuning parameters are needed than in other complex machine learning methods. Logistic regression also performs well when analyzing small datasets and has no overfitting issue. In fact, research [10] has demonstrated that it can do so to estimate treatment effects while also accounting for confounding factors. However, the assumptions accompanying most logistic regression models, such as linearity and multicollinearity, may be violated in real-world data [10]. The algorithm cannot, in the process, model multiplicative complex nonlinear effects or high-order interactions as with tree-based methods. Another disadvantage is that logistic regression is very sensitive to the choice of the classes where the classes are imbalanced. Even if more variables are added to the model, the algorithm may not increase its predictive accuracy if the relationship is not linear or, in other words, if it is nonlinear in nature [11]. These limitations justify the need for superior sampling and variable selection techniques in the study designs. Machine Learning techniques reviewed sources of Linear Regression and Logistic Regression are summarized in Table I.

TABLE I. EVALUATION SUMMARY OF MACHINE LEARNING TECHNIQUES

Ref.	Methodology	Performance metrics	Strengths	Limitations	Result
[7]	Support vector polynomial regression algorithm	Mean absolute error (MAE)	Used number of transactions missing to calculate its performance	Used algorithms may not increase its predictive accuracy and need test it on other algorithms	SVM accuracy: 93%
[8]	Linear regression, basic but fundamental	Mean absolute error (MAE), root mean square error (RMSE)	Simplicity, easy to implement	Limited complexity in modeling	Exceeded 85%
[9]	Regression analysis	MSE, RMSE, R-squared	Practical application	Limited to simple models	Accuracy 76%
[10]	Comparative study of logistic regression and ML	Recall, F1 Score, AUC	Simple and Direct comparison	Algorithm not work as expected with nonlinear models, in addition to that, it is very sensitive to the choice of the classes	N/A
[11]	Logistic regression and Logit boost	Recall, F1 Score, Area Under the Curve (AUC)	Focused on binary classification	The algorithm may not increase its predictive accuracy if the relation is not linear	Accuracy more than 85%

B. Deep Learning

1) Recurrent Neural Networks (RNN): RNNs are a type of deep neural network that is very useful when processing data in a sequence, such as time series data such as stock prices [17]. The recurrent connections allow RNNs to capture temporal dependencies within the series of observations. The study in [18] discussed deep learning stock forecasting models and noted that RNNs/LSTMs can identify temporal dependencies that are not extractible by other models. The inherent recurrent structure of RNNs allows the design of a model of sequential correlations unavailable to feedforward networks. Nonetheless, it is difficult for vanilla RNNs to handle longer sequences because these models have issues with exploding/vanishing gradients [19]. RNNs with memory solve this problem using gated memory cells that help with long-term dependency. In conjunction with recent techniques in deep learning, RNNs offer the best performance in modeling most sequence data. Due to their ability to identify temporal relationships, RNNs with memory models are useful for predicting the next values of time series such as stock prices. RNNs with memory networks for stock price prediction are rooted in enhanced capacity to handle time series data appropriately. The study in [20] compared RNN LSTM with other models, such as linear models and random forests, especially from the perspective of stock price forecasts. The source takes up the problem of a lack of effective tools for risk forecasting in the context of the fluctuating financial market, as well as implementing sound research methods to determine the efficiency of the models [20]. A weakness with all recurrent neural networks is the vanishing gradient problem, but LSTMs are especially lauded for their performance with long-term dependencies in data, which is useful since stock price data are a sequence. This model is even better than regular RNNs because it has memory cells that control the flow of data, and the cells have gates that control the addition and accumulation of data, which tend to solve problems such as vanishing gradients, which are common with regular RNNs.

2) Long Short-Term Memory (LSTM): LSTM is a type of recurrent neural network that might be most effective in modeling time series data. Since their introduction, LSTMs have been implemented in the financial field, where they have been used in stock price prediction, algorithm trading, risk control, and fraud identification. Reference [19] discussed various prediction methods employed in stock markets, including LSTMs. They discovered that LSTMs are more effective for variable financial time series where the data contain nonlinear trends and are nonstationary. They enhance the algorithm to increase its accuracy. Authors of study [21] introduced a four-layer architecture of a dual-LSTM model using past prices and technical indicators to forecast multiple-day-ahead stock prices. The author's approach offers one of the best solutions for benchmark datasets. Accordingly, LSTMs have significant advantages and are suitable for modeling financial time series. It can remember long-range dependencies missing in feedforward networks through their memory cells. Some of the main issues that are particularly relevant to LSTM-based financial applications include data noise, model interpretability, and the model's ability to generalize to distribution shifts with respect to the time series. Research has provided a new mixed option comprising long short-term memory (LSTM) and an adaptive genetic algorithm (AGA) for the prediction of stock indices, further stressing innovation in the spheres of financial forecasting [12]. The study in [22] motioned progressive developments in predictive analytics in financial markets and, thus, considered the current state of the markets. From a methodological perspective, the integration of LSTM with AGA is focused on optimizing the networks' structures and parameters to improve the forecast's accuracy accordingly [22]. Performance metrics such as the mean square and absolute percentage errors are used to check the model's effectiveness. One of the advantages of this study is the authors' effective method of tracking changes in data from one period to another and adjusting for such changes in the analysis, thus enhancing the forecast accuracy. However, the requirements of the components to work out the model result in a more complex design and higher processing needs, which may become an issue of practical application [22]. The results show that it has better accuracy than traditional models do, making it possible to fine-tune investment portfolios [22].

3) CNN-LSTM: The combination of CNNs and LSTM produced a new model that uses both a convolutional neural network (CNN) and LSTM networks to benefit from each other's strengths and make a proper model for time series forecasting. The CNN down samples the input time series to extract local spatial relations, whereas the LSTM learns long-term temporal relations. The CNN is used to extract features, and the output sequence is taken from the CNN and passed to the LSTM at the back end, which is considered for temporal analysis. The CNN features in the network act as translation invariants, and the LSTM capability is used to store the temporal state. A few recent studies have implemented the CNN-LSTM architecture across different fields, such as stock market prediction and electricity load forecasting [20]. Authors of study [23] used a CNN-LSTM to input multivariate financial time series data and established that this improves the performance of a pure LSTM.

4) Analysis of the CNN-LSTM method shows that the CNN-LSTM method for financial time series forecasting takes advantage of convolutional neural networks and long short-term memory networks [23]. It can identify spatial correlations in the data and temporal correlations to make the best prediction. One advantage is that the weights of a CNN are made regular before feeding the LSTM layer, which increases its ability to resist variations [24]. Long-term dependencies may remain a task for a model that can still be challenging compared with more straightforward methods of statistics. Thus, CNN-LSTM has the potential to refine the accuracy of forecasts; however, further studies are still needed in terms of interpretability, versatility for different datasets,

and optimization of feature extraction strategies. Further tweaking and testing it on live data would help achieve its real-life applicability. Simple modifications in the structural design of models and training algorithms could effectively unlock the benefits of this integrated deep learning approach to modeling financial series. Deep Learning techniques reviewed sources of RNN, LSTM, and CNN-LSTM are summarized in Table II.

TABLE II. EVALUATION SUMMARY OF DEEP LEARNING TECHNIQUES

Ref.	Methodology	Performance Metrics	Strengths	Limitations	Result
[21]	Dual-LSTMs, advanced for real-time data	RMSE, Mean Absolute Deviation (MAD)	Capable of handling complex patterns	Complexity in tuning and training	Best average prediction errors of LSTM = 0.05
[19]	Etaheuristic Optimization on RNN–LSTM	MAE, R-squared, Accuracy, Precision	Wide scope by using more datasets	Lacks specific practical insights	Accuracy after using RNN–LSTM increment of 4–6%
[23]	Multivariate CNN-LSTM model	MSE, RMSE, MAE	Handles multiple data streams	Complexity and implementation difficulty	CNN-LSTM Average value of RMSE = 0.0162
[22]	Uses LSTM with sentiment analysis from social media. (TLSTM)	RMSE, MAE, Accuracy, Precision	Combines technical analysis with sentiment analysis for better predictions	Model Complexity	RMSE= 2.147, 82.19 and F-measure = 89%

IV. METHODOLOGY

A. Research Design and Framework

The study employs a quantitative experimental design to determine the performance of machine learning and deep learning techniques for stock market prediction. The methodology outlines the procedure as a six phase process: data collection, preprocessing, models implementation, model training/testing, and results evaluation. This study will only consider the performance of the Saudi Aramco (2222.SR) stock. Saudi Aramco is the largest oil company in the world, and it has considerable influence over the regional and global financial markets [1].

The experimental framework was structured to allow reproducibility and statistical reliability due to a comparison of three modeling approaches. The first Linear Regression model was used as a baseline model. The second, RNN, was a first-generation model based on basic sequential learning. The third, LSTM, was a first-generation model based on sequential temporal dependencies. This multi-model approach allows for assessing performance at different levels of complexity and computational overhead. In addition, it provides a better understanding of how machine learning methods perform compared to the more up-to-date deep learning architectures in forecasting applications within finance.

B. Data Collection and Acquisition

Historical stock data for Saudi Aramco was collected systematically using Yahoo Finance API (yfinance) Python library over a total of last five years. The dataset captures daily trading activity which includes opening price, closing price, high-low range for the day, volume of trades and is therefore complete in terms of transaction history. Since the target stock symbol was 2222.SR, it ensured that the data collected and returned was specifically from the Saudi Stock Exchange (Tadawul). Generally, error handling and data validation were built into the data collection process to capture concerns regarding completeness and integrity of the data returned. Missing values were identified and subsequently addressed using forward and backward interpolation as appropriate, when generating daily averages from weekly averages. Outliers were identified through statistical measures and determined whether to keep/discard outliers based on trading volumes and more importantly, where the trades were made with respect to the market, bid-ask and consensus price at that point in time. Overall, the comprehensive dataset provides a great way to train a model, and based on the extensive and complete dataset provided, all empirical results should reflect the actual market conditions and trading behaviors observed in the Saudi Arabian equities market.

C. Preprocessing

The preprocessing stage included many data standardization methods to improve how well the model converges with optimal performance. The data standardization techniques included standardization of the input features utilizing Z-score normalization method to standardize all input features using the equation, as shown in Eq. (1):

$$Z = (X - \mu) / \sigma \quad (1)$$

Where Z is the standardized value of X, the original value, μ is the mean, and σ is the standard deviation of the feature. The standardization of the input features is important in optimizing a neural network as it allows the converging of the cost function to occur more efficiently [14]. This ultimately reduces the required number of epochs to train the model and improves model reliability. The preprocessing pipeline developed the temporal sequences for the sliding window techniques in RNN and LSTM models. The construction of temporal sequence input patterns using sliding windows, enabled the models to learn from historical price movements. All numerical features underwent feature scaling to develop an ability to separate features within differing magnitude therefore preventing bias to larger magnitude features. The dataset was divided into a training and testing dataset, the training dataset consisted of 80% and the testing dataset consisted of 20% of the total data, used chronological splitting methods to prevent any leakage of future data which ensured the temporal characteristics of the dataset remained intact throughout the inception of the models.

D. Model Performance Evaluation Metrics

The model performance was assessed using two main evaluation metrics. The first evaluation metric was Root Mean Square Error (RMSE), which was calculated as shown in Eq. (2):

$$RMSE = \sqrt{(\Sigma(y_i - \hat{y}_i)^2 / n} \quad (2)$$

The second evaluation metric was Mean Absolute Error (MAE), calculated as shown in Eq. (3):

$$MAE = \Sigma|y_i - \hat{y}_i| / n \quad (3)$$

where y_i represents the actual values, $\hat{y}_i$ represents the predicted values, and n is the number of observations that were captured each day. After evaluating all models, meaningful differences in predictive performance were observed across models, with results systematically recorded to analyze prior to determining the best approach for predictive modeling.

E. Model Implementation and Architecture

The Linear Regression model is the baseline with which everything else can be compared. It implements the simplest relationship in the literature, which is shown in Eq. (4):

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_n X_n + \varepsilon, \quad (4)$$

has easy-to-interpret coefficients, and is easy to compare with this work. The Linear Regression model was implemented using scikit-learn's Linear Regression class with all parameters set to the default (that is, no user-supplied input / adjusted parameters for this baseline model) to standardize the evaluation conditions. The implementation of the RNN architecture was carried out using the TensorFlow/Keras framework. It defines sequential data with the appropriate number of timesteps in the input layer, an RNN layer that has 50 hidden units, a dense layer that contains the predicted closing price, an Adam optimizer with a default learning rate set to 0.001, and finally a Mean Squared Error loss function. The RNN model used a basic architecture which was capable of looking at (and capturing) some short-range temporal dependencies, which may still have inherent issues with long range dependencies due to vanishing gradient issues before the learner reaches the next timestep [21]. The LSTM structure uses memory cells and gating mechanisms to solve RNN issues. It has an LSTM layer containing 50 memory units, a dropout layer (0.2) to regularize, and a dense final output layer for prediction. It also has the same optimizer and loss function as RNN. The LSTM design allows for learning long-term dependencies using a more sophisticated gating architecture, including forget gates, input gates, and output gates [21].

F. Statistical Validation, Robustness Testing, and Computational Environment

This study employed a rigorous method to implement and test machine learning models, including linear regression, with machine learning versus strictly statistical. The intention was to determine models that yield the most stable predictive accuracy to be used specifically in the Saudi Arabian equity market. Cross-validation tested the generalization capacity of each model by maintaining training and test data distinct to avoid leakage and attain a realistic performance estimation. Sensitivity analysis tested model stability against small to moderate perturbations in hyperparameters. Statistical significance testing determined whether differences observed in performance were statistically significant, for example, linear regression versus other models. Robustness testing assessed the performance of models under different market conditions and volatility regimes to assess generalizability. The performance of the models were assessed using the same computational resources; all experiments used Python 3.8+ with the same libraries (TensorFlow 2.x, scikit-learn, pandas, and numpy). Random seeds were established and hyperparameter values were recorded as well as calculated to ensure reproducibility. Each model had the same hardware specifications, details of all training modalities were finalized and recorded to allow full transparency and facilitate future model improvements.

V. Results and Evaluation

The investigation findings provide a comparative evaluation among three predictive methods for forecasting Saudi Aramco stock prices: Linear Regression, Long Short-Term Memory (LSTM), and Recurrent Neural Network (RNN). The training and testing datasets were evaluated using two relevant regression measures: Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE). In the training phase, Linear Regression was again the best-performing method with the lowest RMSE (0.334) and MAE (0.203). For these metrics, linear regression was performed comparably on a direct comparison basis with the actual price. LSTM at RMSE 0.355 and MAE 0.266 for the remaining predictive methods, and RNN at RMSE 0.383 and MAE 0.263. Training Results prediction performance for Aramco Stock Market is summarized in Table III.

Thus, although the RNN and LSTM are deep learning computational models that model temporal dynamics such as time series data and patterns over time, and deeper network connections will provide better mathematical representation, they did not outperform the Linear regression-based training performance.

More importantly, during the testing phase, which indicates how the model performs on unseen data, Linear Regression (LR) remained ahead of the pack, with an RMSE of 0.304 and MAE of 0.207, as shown in Table IV and drawn in Fig. 2. LSTM's RMSE was 0.346 and MAE of 0.237, as shown in Table IV and drawn in Fig. 3, while RNN performed the worst with an RMSE of 0.415 and MAE of 0.329, as shown in Table IV and drawn in Fig. 4. Testing results analysis for Aramco stock market are summarized in Table IV.

TABLE III. Training Results Prediction Performance of Aramco Stock Market

Model Name	RMSE	MAE
Linear Regression	0.334	0.203
LSTM	0.355	0.266
RNN	0.383	0.263

TABLE IV. Testing Results Analysis of Aramco Stock Market

Model Name	RMSE	MAE
Linear Regression	0.304	0.207
LSTM	0.346	0.237
RNN	0.415	0.329

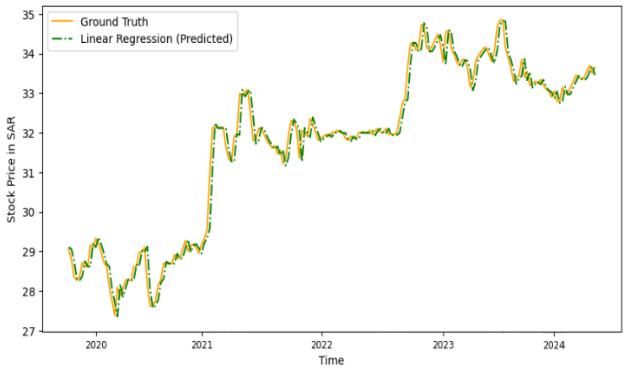


Fig. 2. LR prediction performance of Aramco stock market.

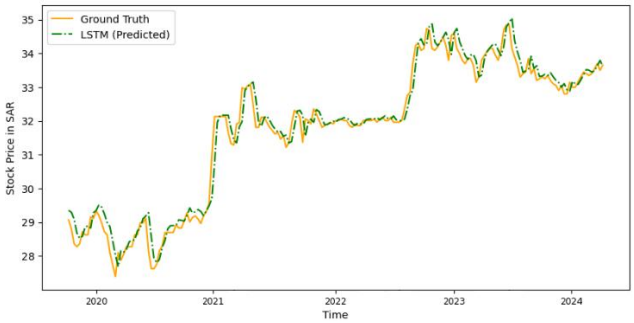


Fig. 3. LSTM prediction performance of Aramco stock market.

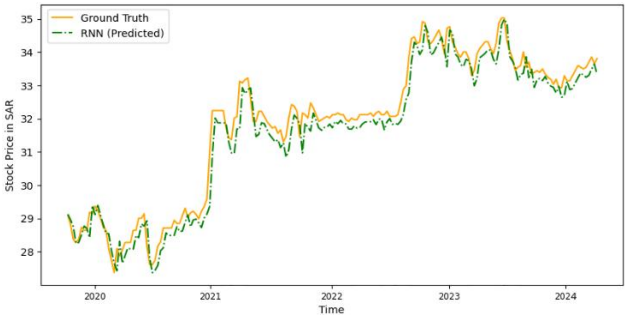


Fig. 4. RNN prediction performance of Aramco stock market.

A. Discussion

The study evaluated three specific models: Linear Regression (LR) from Machine learning, Recurrent Neural Networks (RNNs), and Long Short-Term Memory (LSTM) networks from Deep learning.

1) Linear Regression (LR): The study found that LR performed better than the deep learning models. This demonstrates that a less complex model can offer better forecasting accuracy, which is a significant finding. However, LR is limited in its ability to solve very complicated problems.

2) Long Short-Term Memory (LSTM) Networks: LSTMs are an advanced form of RNNs that address the vanishing gradient problem by using memory cells. The study found that LSTMs had a higher RMSE error compared to LR and performed better than RNNs. Its issues in this experiment are related to its complexity.

3) Recurrent Neural Networks (RNNs): The study's evaluation showed RNNs with the highest RMSE. A key weakness of standard RNNs is their difficulty in handling longer sequences due to issues with exploding modeling time series data, especially with non-linear and non-stationary trends.

Ultimately, this indicates that when assumed as a machine learning model and thus trained and treated as such, Linear Regression will outperform more complex Deep learning models - like RNNs and LSTMs - on stock market prediction tasks particularly when sequential patterns are not overly prevalent in the data or when the dataset is smaller in size. In conclusion, the findings continue to emphasize how model selection should be based on understanding the data. DNNs can include hidden layers meant to learn complex patterns in data, but models (assumptions of the data/structure) can still reveal the best predictive accuracy. While there is an excellent opportunity for deep learning models, their performance is not always superior, particularly in cases where the simpler model reveals the best outcomes.

VI. Critical Analysis and Synthesis of Key Findings, Gaps, and Trends

A central finding of the research is that a simpler model like Linear Regression outperformed more complex deep learning models (RNN and LSTM) in terms of lower error rates. This contradicts the general trend in machine learning, where deep learning models are often lauded for their superior ability to handle complex, non-linear data. This finding suggests that while DL models are powerful, their effectiveness in stock market prediction is not guaranteed and is highly context-sensitive and data-driven. It could be that the specific dataset or preprocessing methods used in the study were more suited to a linear model, or that the deep models were not sufficiently optimized.

A. The Dominance of Deep Learning for Non-linear Data

Despite the specific findings of the study, the broader literature review consistently highlights the shift from traditional statistical methods to more sophisticated AI technologies (ML and DL) to analyze the market's complex and non-linear data. Deep learning models, particularly LSTMs, are specifically praised for their capacity to handle time series data and mitigate the vanishing gradient problem, making them suitable for the long-term dependencies inherent in financial data. The trend is also moving toward hybrid models, like CNN-LSTM, which aim to combine the strengths of different architectures to improve prediction accuracy by capturing both spatial and temporal features.

B. Data-Centric and Preprocessing Trends

The study explicitly mentions the use of "modern data preprocessing procedures," including Z-score normalization and temporal sequence, to ensure effective model training. The literature also notes the increasing use of large, heterogeneous datasets that go beyond historical prices to include technical indicators, financial news, and social media sentiment. This trend towards incorporating diverse data sources is a key advancement, as it enriches the models with more context and helps them adapt to changing market conditions.

C. Gaps and Underexplored Areas

The study identifies several limitations and areas for future research:

1) Model interpretability: While models like Linear Regression offer interpretability, deep learning models often have complex functions. The paper notes that model interpretability is a key issue, especially for LSTM-based financial applications.

2) Generalizability and robustness: The study highlights that a model's ability to generalize to "distribution shifts" over time is a significant issue. The non-stationary nature of stock data means models must be periodically retrained on recent data.

3) The "Human Element": This study mentions incorporating predictions and insights into trading systems for "prescriptive analytics" and "integration with human decision-making". This suggests a gap in current research, which often focuses solely on model accuracy, without fully exploring the integration of these AI systems with human expertise and trading strategies.

4) Real-world applicability: The paper mentions that further tweaking and testing on "live data" is needed to achieve real-life applicability for models like CNN-LSTM. This points to a gap between theoretical model performance and practical, real-world deployment.

The findings of this research extend beyond the performance of individual models, offering insights into core, ongoing challenges in computer science and its applications.

1) The most striking finding, where the simple Linear Regression (LR) model outperformed the more complex deep learning models (RNNs and LSTMs), the study provides experimental evidence that for financial forecasting, which is both context-sensitive and data-driven, a complex model is not always the optimal choice. This highlights the ongoing challenge for computer scientists to select, not just develop, the right algorithm for a specific problem and dataset rather than universally favoring the latest or most sophisticated architecture.

2) Adaptive systems for dynamic environments: The document repeatedly mentions the challenge of nonstationarity in stock data and the need to periodically retrain models on recent data. This is a central problem in computer science applications that deal with dynamic, evolving systems, from robotics to network security. The need for models to adapt to distribution shifts over time without manual intervention is a major research area. The findings from this study confirm that for financial data, which is highly sensitive to real-world events and changes, the ability of a model to generalize and remain robust is a critical, and often unfulfilled, requirement.

3) Hybrid architectures for complex problems: The research's focus on hybrid models like the CNN-LSTM directly reflects a major trend in modern computer science: the move toward multi-modal or hybrid architectures. By combining different types of neural networks, researchers aim to leverage the unique strengths of each component to solve more complex problems than a single model could alone. The CNN-LSTM architecture, for example, is designed to extract both spatial features and temporal relationships. This approach represents a shift from finding a single "best" algorithm to building composite systems that are better suited for the multifaceted nature of real-world data.

VII. Conclusion

From a practical perspective, the research has important guidance to provide traders and financial analysts on linear and non-linear models. A linear model has features such as interpretability, rapid calculation speed, and reliable performance that make it a practical choice for real-time trading systems. The work did reveal that some models used in the experiments produced reliable predictions in empirical tests and supported context-based model selection.

This research investigated the capacity of machine learning and deep learning models to predict prices for the consumed stock of Saudi Aramco. The results demonstrated that the simpler models such as the Linear Regression model were able to achieve lower RMSE and MAE errors than LSTM and RNN models. This suggests that simpler models can capture stock movements, which seem to be relatively linear. The lower performance of the deep-learning models suggests that they likely overfit the data, leading to the notion that not all complexities lead to better forecasting models. This study highlighted the relevance of regression models in financial forecasting. This is also highly applicable to emerging markets, where data characteristics are not conducive to complex neural architectures.

Future work should apply to a wider diversity of stocks, incorporate outside economic variables, and investigate ensemble models containing both linear and deep learning components. Future research also needs to focus on making complex models more transparent so that implementers can understand why a particular prediction was made. Further research is needed to create models that are more robust and adaptable to sudden and unexpected market changes. This is crucial for building trust and enabling human decision-making. All these then provide the basis for smarter and more efficient specific decision-making in algorithmic trading, risk management, and portfolio optimization

References

[1] Y. C. Chen and W. C. Huang, “Constructing a stock-price forecast CNN model with gold and crude oil indicators,” *Appl. Soft Comput.*, vol. 112, p. 107760, Jan. 2021.

[2] Li, X., Wu, P., & Wang, W., “Incorporating stock prices and news sentiments for stock market prediction: A case of Hong Kong,” *Inf. Process. Manag.*, vol. 57, no. 5, 102212, Sep. 2020.

[3] A. M. Pattanayak, A. Swetapadma, and B. Sahoo, “Exploring different dynamics of recurrent neural network methods for stock market prediction—A comparative study,” *Appl. Artif. Intell.*, vol. 38, no. 1, 2371706, Jan. 2024.

[4] F. M. P. Fozap, “Hybrid machine learning models for long-term stock market forecasting: Integrating technical indicators,” *J. Risk Financial Manag.*, vol. 18, no. 4, p. 201, Apr. 2025.

[5] J. Ayala, M. García-Torres, J. L. V. Noguera, F. Gómez-Vela, and F. Divina, “Technical analysis strategy optimization using a machine

learning approach in stock market indices," *Knowl.-Based Syst.*, vol. 225, 107119, Jul. 2021.

[6] D. R. Khan, A. B. Patankar, and A. Khan, "An experimental comparison of classic statistical techniques on univariate time series forecasting," *Procedia Comput. Sci.*, vol. 235, pp. 2730–2740, Apr. 2024.

[7] E. Gambhir, R. Jain, A. Gupta, and U. Tomer, "Regression analysis of COVID-19 using machine learning algorithms," in *Proc. Int. Conf. Smart Electron. Commun. (ICOSEC)*, 2020, pp. 65–71.

[8] P. Yu and X. Yan, "Stock price prediction based on deep neural networks," *Neural Comput. Appl.*, vol. 32, no. 6, pp. 1609–1628, Jun. 2020.

[9] C. Thakar and S. Ratnaparkhi, "Machine learning – Predict stock prices using regression," *Quantitative Finance & Algo Trading Blog by QuantInsti*, Apr. 18, 2023. [Online]. Available: https://blog.quantinsti.com/machine-learning-trading-predict-stock-prices-regression/

[10] S. Nusinovici, Y. C. Tham, M. Y. C. Yan, D. S. W. Ting, J. Li, C. Sabanayagam, T. Y. Wong, and C. Y. Cheng, "Logistic regression was as good as machine learning for predicting major chronic diseases," *J. Clin. Epidemiol.*, vol. 122, pp. 56–69, Jun. 2020.

[11] H. Jain, A. Khunteta, and S. Srivastava, "Churn prediction in telecommunication using logistic regression and logit boost," *Procedia Comput. Sci.*, vol. 167, pp. 101–112, Mar. 2020.

[12] I. Paliari, A. Karanikola, and S. Kotsiantis, "A comparison of the optimized LSTM, XGBOOST and ARIMA in time series forecasting," in *Proc. Int. Conf. Inf., Intell., Syst. Appl. (IISA)*, 2021, pp. 1–7.

[13] A. O. Aseeri, "Effective RNN-based forecasting methodology design for improving short-term power load forecasts: Application to large-scale power-grid time series," *J. Comput. Sci.*, vol. 68, p. 101984, Mar. 2023.

[14] D. K. Mademlis and N. Dritsakis, "Volatility forecasting using hybrid GARCH neural network models: The case of the Italian stock market," *Int. J. Econ. Financial Issues*, vol. 11, no. 1, pp. 49–60, 2021.

[15] L. Rubio, A. P. Pinedo, A. M. Castaño, and F. Ramos, "Forecasting volatility by using wavelet transform, ARIMA and GARCH models," *Eurasian Econ. Rev.*, vol. 13, no. 3–4, pp. 803–830, 2023.

[16] B. Lin and T. Lan, "The time-varying volatility spillover effects between China's coal and metal market," *J. Futures Mark.*, vol. 44, no. 5, pp. 699–719, May 2024.

[17] R. Gomila, "Logistic or linear? Estimating causal effects of experimental treatments on binary outcomes using regression analysis," *J. Exp. Psychol. Gen.*, vol. 150, no. 4, pp. 700–709, Apr. 2021.

[18] W. Jiang, "Applications of deep learning in stock market prediction: Recent progress," *Expert Syst. Appl.*, vol. 184, 115537, Dec. 2021.

[19] K. Kumar and M. T. U. Haider, "Enhanced prediction of intra-day stock market using metaheuristic optimization on RNN–LSTM network," *New Gener. Comput.*, vol. 39, no. 1, pp. 231–272, Jan. 2021.

[20] W. Lu, J. Li, Y. Li, A. Sun, and J. Wang, "A CNN-LSTM-based model to forecast stock prices," *Complexity*, vol. 2020, pp. 1–10, Aug. 2020.

[21] K. Gajamannage, Y. Park, and D. I. Jayathilake, "Real-time forecasting of time series in financial markets using sequentially trained dual-LSTMs," *Expert Syst. Appl.*, vol. 223, 119879, Jan. 2023.

[22] A. Peivandizadeh, S. Hatami, A. Nakhjavani, L. Khoshsima, M. R. C. Qazani, M. Haleem, and R. Alizadehsani, "Stock market prediction with transductive long short-term memory and social media sentiment analysis," *IEEE Access*, vol. 12, pp. 87110–87130, 2024.

[23] H. Widiputra, A. Mailangkay, and E. Gautama, "Multivariate CNN-LSTM model for multiple parallel financial time-series prediction," *Complexity*, vol. 2021, no. 1, 9903518, 2021.

[24] Ş. Öztürk and U. Özkaya, "Residual LSTM layered CNN for classification of gastrointestinal tract diseases," *J. Biomed. Inform.*, vol. 113, 103638, Jan. 2021.

Consumer Adoption of Autonomous Vehicles in China: A Bibliometric Review of Intention Drivers and Perceptions

Yunluo Zou[1]*, Syuhaily Osman[2], Sharifah Azizah Haron[3]
Faculty of Human Ecology, Universiti Putra Malaysia, Selangor, Malaysia[1, 2, 3]
College of Economics & Management, Hubei University of Arts and Science, Xiangyang, China[1]

***Abstract*—Autonomous vehicles (AVs) are playing an increasing role in digitally enabled transportation systems with the dramatic emergence of related technologies. Consumer adoption is arguably a key factor in the deployment of AVs in China. In this study, bibliometric analysis was used to explore intention drivers regarding consumer adoption of AVs and the role of consumer perception in the decision-making process from the perspective of Chinese consumers. The results revealed that consumer perception is a highly critical factor influencing the adoption of AVs. Moreover, most Chinese consumers were more sensitive to perceived losses than to gains. In addition, the main public focus was on highly intelligent shared AVs rather than family-use vehicles. These findings could help governments and enterprises gain a deeper understanding of consumer behavior in the Chinese market, which could be used as a reference for implementing measures to better accelerate the diffusion of AVs.**

Keywords*—Autonomous vehicles; diffusion of AVs; consumer adoption; consumer perception; Chinese market; bibliometric analysis

I. Introduction

Autonomous vehicles (AVs) have become regarded as one of the most active and promising innovations in the global market in recent years [1]. AVs are vehicles with the capability to guide themselves without human control [2]. The Society of Automotive Engineers (SAE) classifies autonomous vehicles into six distinct levels of driving automation, ranging from none to full automation. At Levels 4 and 5, these vehicles are driven almost entirely by AI, a process also known as unmanned driving [3]. Although AV-related industries and technologies are still developing and at their initial stage, the application of these vehicles is regarded as having considerable potential. Governments and the public believe these vehicles can perform successfully in promoting safety, reducing energy consumption, and improving traffic efficiency [4]. For example, the World Economic Forum (WEF) predicted that AVs would shape future urban mobility [5].

It is believed that China is experiencing one of the most rapid diffusion of AVs worldwide, with the expectation being that around 250,000 AVs equipped with Level 4 intelligence will be sold in 2034 [6]. Undoubtedly, consumer adoption is an important variable influencing this diffusion. Scholars have found that consumer adoption of AVs is driven by several factors. However, these factors may vary across cultures, market conditions, and policies. For example, studies in the USA found that environmental concerns, safety concerns, and costs contributed most to consumer attitudes towards AVs [7]. Therefore, to accelerate the diffusion of AVs in China, this study aims to identify the factors influencing consumer adoption intention and the perception-driven decision-making mechanisms behind AV adoption.

Although the Chinese AV market has attracted extensive academic research, this has primarily focused on technology development. Relatively few articles have conducted literature reviews from the perspective of consumers. Thus, the current study can bridge this gap and help determine what consumers are thinking and pursuing, as well as what they hate. It could also provide practical methods to accelerate the diffusion of this technology in China.

This study is organized as follows: In Section II, the development status of the Chinese AV market, consumer adoption intention, factors influencing consumer intention, and decision-making mechanisms involving consumer perception are presented based on the existing literature. The research questions of this study are then proposed. In Section III, the research methodology, data sources, search strategies, and screening and inclusion criteria are introduced. In Section IV and Section V, the results of two bibliometric analyses on factors influencing consumer intention to adopt AVs and consumer perception of AVs are presented and discussed. In Section VI, conclusions and implications are provided.

II. Literature Review and Research Questions

A. Development of Autonomous Vehicles in China

In 2019, McKinsey & Company [8] predicted that China would likely become the world's largest autonomous vehicle market and that two trillion dollars would have been generated in this market by 2040.

In the timeline formulated by McKinsey for the Chinese autonomous vehicle market, the adoption of Level 4 AV technology would start in 2027. In reality, this market has developed at a speed exceeding McKinsey's expectations. In 2024, China embarked on numerous pilot programs involving unmanned buses and taxis equipped with Level 4 intelligence [9].

Apart from its application in shared mobility, the automation of passenger vehicles is transitioning from Level 2

*Corresponding Author.

(partial automation) to Level 3+ (conditional automation). According to a Counterpoint report in 2024 [10], China is going to become the primary driver in enhancing machine intelligence levels, with the number of its Level 3 passenger vehicles expected to exceed one million by 2026.

In the field of logistics, the unmanned delivery business in China is developing rapidly. During the Double 11 Shopping Festival in 2023, unmanned delivery vehicles equipped with L4 autonomous driving technology were introduced in multiple cities, such as Hangzhou, delivering around 800 parcels per car each day [11].

Based on a Renub Research report in 2025 [12], the Chinese autonomous vehicle industry is exhibiting a rapidly upward trend. In 2024, the AV market size reached 17.23 billion dollars. With a CAGR of 29.01%, it is predicted that the market size will increase to 170.57 billion dollars by 2033, as shown in Fig. 1.

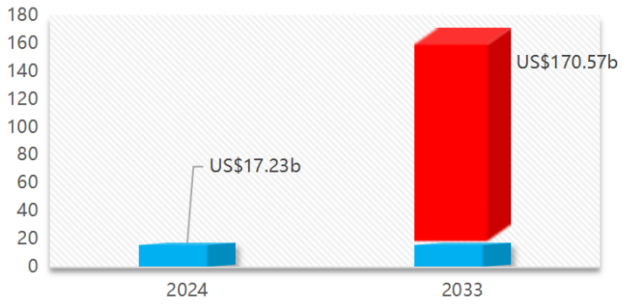


Fig. 1. Market size of autonomous vehicles in China (source: Renub Research).

B. Consumer Adoption Intention in the Chinese AV Market

Since autonomous vehicles are not yet widespread, it is challenging to obtain accurate and continuous data about the adoption intentions of consumers. However, by utilizing various commercial marketing company reports, it is possible to gain a basic understanding of consumers' attitudes in China.

Ipsos surveys from 2018 to 2022 [13], [14] demonstrated an obvious trend of increasing acceptance of autonomous vehicles in Asian countries compared to Western countries. Set against the global average, Chinese people have a higher intention to adopt autonomous vehicles. However, this intention is not exhibiting a significant upward trend.

According to these reports, the percentage of consumers in China who were willing to buy autonomous vehicles rose slightly from 46% to 51% between 2018 and 2022. This phenomenon is not an isolated case across Asia. In Malaysia, the proportion was 48% in 2018, so this country ranked second among all those surveyed. However, in 2020, only 47% of Malaysians believed that AVs would be a part of everyday life [15], as shown in Fig. 2.

C. Factors Influencing Consumer Intention to Adopt AVs

Market expectations appear to be far greater than the speed at which consumer willingness is increasing. Therefore, certain reasons must be limiting the growth of consumer adoption of AVs in China. Scholars have found that consumer adoption of AVs is influenced by many factors, including internal and external aspects.

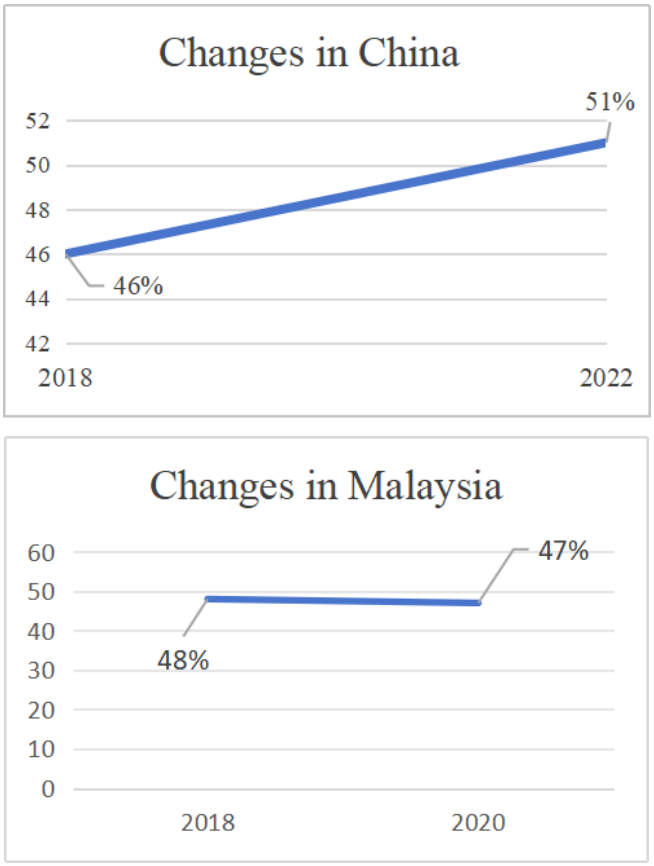


Fig. 2. Changes of intention in China and Malaysia (Source: Ipsos).

Topol š ek, Babić, Babić, and Cvahte Ojster š ek [16] revealed that the widespread adoption of autonomous vehicles was inevitable. Their research suggested that a series of factors influence consumer adoption intention of AVs, including safety, age, education levels, social influence, and consumer perceptions. Jing, Xu, Chen, Shi, and Zhan [17] investigated and compared the status of Europe and Asia, discovering that 13 factors, such as safety, influence acceptance of autonomous vehicles. On the other hand, Islam, Abdel-Aty, Lee, Wu, Yue, and Cai [18] showed that demographic traits appear to have no significant effect on consumer adoption intentions related to AVs.

A series of studies have revealed that consumer perception can explain how consumers make adoption decisions in the autonomous vehicle market. Chikaraishi, Khan, Yasuda, and Fujiwara [19] conducted a survey of public perceptions of AVs in Japan, finding that residents cared significantly more about perceived gains rather than losses in this market. This perception mechanism had obviously influenced AV acceptance in Japan. Meanwhile, Cai and Xiong [20] focused on AV-related privacy issues, revealing that when consumers faced these issues, they usually made a trade-off between perceived safety and perceived privacy risk. The scholars discovered that if consumers believed they were sharing data to improve safety, they would perceive more benefits than risks.

D. Decision-Making Mechanism with Consumer Perception

Scholars have developed a series of theories to explain the decision-making mechanisms with perceptions. Perceived Value Theory explores how humans determine their behavior based on the perceived value of goods or services, which is influenced by perceived gains and costs [21]. Bauer proposed Perceived Risk Theory, whose core idea is that any purchasing behavior involves uncertainty that makes consumers feel negatively, such as experiencing anxiety, worry, or discomfort, because individuals cannot predict the consequences of their decisions [22]. This theory focuses on the negative perceptions of consumers and how they function. The Value Acceptance Model (VAM) proposed by Kim, Chan, and Gupta clearly explains the decision-making mechanism as a trade-off between positive and negative perceptions. Its core concept is perceived value, which is determined by positive and negative perceptions [23], as shown in Fig. 3. These have been classified as perceived benefits and perceived sacrifices. The former include usefulness and enjoyment, while the latter include technicalities and fees.

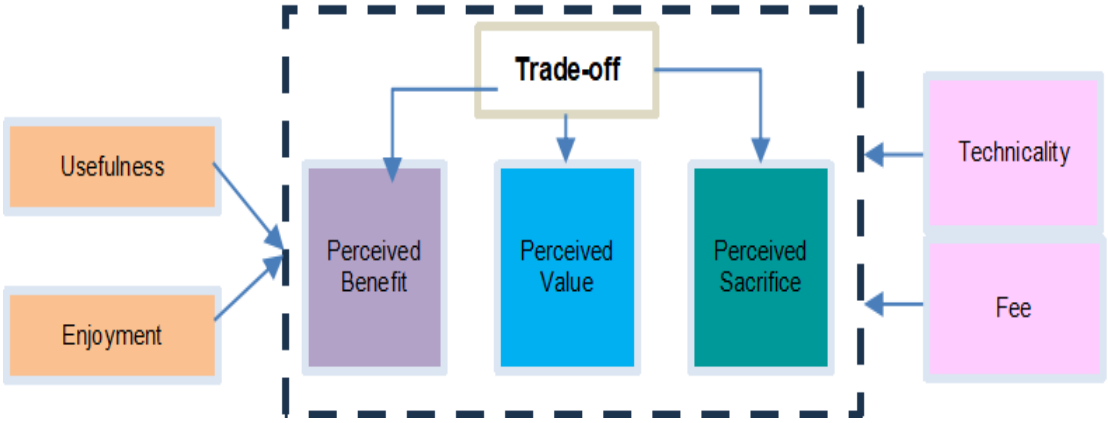


Fig. 3. Schematic representation of the VAM model.

To discover the decision-making mechanism of Chinese consumers in regard to autonomous vehicles, a deeper understanding of their perceptions is necessary. To explore the perceptions of these consumers, a bibliometric analysis was conducted.

E. Contributions of the Paper and Research Questions

The objectives of this study are to identify the intention of drivers regarding consumer adoption of AVs and understand the decision-making mechanisms behind AV adoption in China from the perspective of consumer perceptions, which would be achieved through bibliometric analysis. The study should enhance the understanding of Chinese consumer behavior within the government and enterprises, which can then propose better strategies to accelerate the diffusion of AVs.

Based on the research objective, the following research questions were proposed:

- What factors influence consumer intention to adopt AVs in China?
- In China, what are the consumer perceptions of autonomous vehicles?
- How can these perceptions influence the decision-making process of consumers?

III. Methodology

To determine the influencing factors of adoption intention and consumer perceptions of AVs in China, two bibliometric analyses were conducted using Bibliometrix [24].

A. Sources and Search Strategy

The data for both studies were extracted from the SCOPUS and IEEE Xplore databases, covering a 10-year period from 2016 to 2025. The following search keywords and rules were established for use in SCOPUS and IEEE Xplore, as shown in Fig. 4 and Fig. 5 for the two bibliometric analyses. The search string and keywords from Fig. 4 were applied to the bibliometric analysis of factors influencing consumer intention to adopt AVs in China; likewise, the search string and keywords from Fig. 5 were applied to the bibliometric analysis of consumer perception of AVs.

Database: Scopus

Search: (TITLE-ABS-KEY("autonomous vehicle*") OR (TITLE-ABS-KEY("autonomous car*") OR TITLE-ABS-KEY("driveless vehicle*") OR TITLE-ABS-KEY("driveless car*") OR TITLE-ABS-KEY("self-driving vehicle*") OR TITLE-ABS-KEY("self-driving car*")) AND (TITLE-ABS-KEY(intention) OR TITLE-ABS-KEY(will*) OR TITLE-ABS-KEY(accept*) OR TITLE-ABS-KEY(deci*))AND (TITLE-ABS-KEY("factor*") OR TITLE-ABS-KEY(influen*)) AND (TITLE-ABS-KEY(China) OR TITLE-ABS-KEY(Chinese) OR AFFILCOUNTRY(China)))

Database: IEEE Xplore

Search: ("All Metadata":China OR "All Metadata":Chinese) AND ("Abstract":"autonomous vehicle" OR "Abstract":"autonomous vehicles" OR "Abstract": "autonomous car" OR "Abstract":"autonomous cars" OR "Abstract": "self-driving vehicle" OR "Abstract":"self-driving vehicles" OR "Abstract": "self-driving car" OR "Abstract":"self-driving cars" OR "Abstract": "driverless vehicle" OR "Abstract":"driverless vehicles" OR "Abstract": "driverless car" OR "Abstract":"driverless cars" OR "Abstract": "unmanned vehicle" OR "Abstract":"unmanned vehicles" OR "Abstract": "unmanned car" OR "Abstract":"unmanned cars") AND ("Abstract":intention OR "Abstract":intentions OR "Abstract":willingness OR "Abstract":will OR "Abstract":accept OR "Abstract":acceptance OR "Abstract":decision OR "Abstract":decisions OR "Abstract":decide OR "Abstract":adoption OR "Abstract":adopt) AND ("Abstract":factor OR "Abstract":factors OR "Abstract":influence OR "Abstract":influences OR "Abstract":influencing)

Fig. 4. Search keywords and rules in databases for factors influencing intention to adopt AVs

Database: Scopus

Search: (TITLE-ABS-KEY ("autonomous vehicle*") OR (TITLE-ABS-KEY ("autonomous car*") OR TITLE-ABS-KEY ("driveless vehicle*") OR TITLE-ABS-KEY ("driveless car*") OR TITLE-ABS-KEY ("self-driving vehicle*") OR TITLE-ABS-KEY ("self-driving car*")) AND (TITLE-ABS-KEY (perceive*) OR TITLE-ABS-KEY (perception*)) AND (TITLE-ABS-KEY (user*) OR TITLE-ABS-KEY (consume*) OR TITLE-ABS-KEY (customer*) OR TITLE-ABS-KEY (driver*) OR TITLE-ABS-KEY (passenger*) OR TITLE-ABS-KEY (public*)) AND (TITLE-ABS-KEY (china) OR TITLE-ABS-KEY (chinese) OR AFFILCOUNTRY (china)))

Database: IEEE Xplore

Search: ("All Metadata":China OR "All Metadata":Chinese) AND ("Abstract":perception OR "Abstract":perceptions OR "Abstract":perceive OR "Abstract":perceived OR "Abstract":perceives) AND ("Abstract":user OR "Abstract":users OR "Abstract":consumer OR "Abstract":consumers OR "Abstract":customer OR "Abstract":customers OR "Abstract": driver OR "Abstract":drivers OR "Abstract":passenger OR "Abstract":passengers OR "Abstract":public) AND ("Abstract":"autonomous vehicle" OR "Abstract":"autonomous vehicles" OR "Abstract":"autonomous car" OR "Abstract":"autonomous cars" OR "Abstract": "self-driving vehicle" OR "Abstract":"self-driving vehicles" OR "Abstract":"self-driving car" OR "Abstract":"self-driving cars" OR "Abstract": "driverless vehicle" OR "Abstract":"driverless vehicles" OR "Abstract":"driverless car" OR "Abstract":"driverless cars")

Fig. 5. Search keywords and rules in databases for consumer perception.

B. Screening and Inclusion Criteria

To ensure the selected literature focused specifically on: 1) factors influencing consumer adoption intention regarding AVs and 2) consumer perceptions, with all the studies conducted in the Chinese context, the PRISMA model was employed for systematic literature screening. The final analysis included 107 and 87 qualified publications for Study 1 and Study 2, respectively.

C. Process for Extracting Records from Databases for Study 1

Initial searches yielded 1,027 publications, with 800 from SCOPUS and 227 from IEEE Xplore. To perform a preliminary screening of the literature, restrictions were applied based on the publication year, research area, document type, and language. Only studies published in English between 2016 and 2025 and within specific document types were included. Similarly, by limiting the research areas, studies that were highly relevant to the review topic were included. According to the inclusion criteria, the number of studies was 661 from SCOPUS and 192 from IEEE Xplore. The process and results are shown in Table I.

TABLE I. PRELIMINARY SCREENING OF LITERATURE FOR STUDY 1

Filter	SCOPUS	IEEE Xplore
Publication Year	2016-2025 (n = 800)	2016-2025 (n = 227)
Research Area	Computer science; Social sciences; Engineering; Business, Management and accounting; Decision sciences; Economics, econometrics and finance; Psychology (n = 744)	Autonomous Vehicles; Human Drivers; Self-driving; Automated Vehicles (n = 192)
Document Type	Article; Conference paper; Review; Book chapter (n = 739)	Journals ; Conferences; Early Access Articles; Magazines (n = 192)
Language	English (n = 661)	English (n = 192)

To further filter out less appropriate literature, the preferred reporting items for systematic reviews and meta-analyses (PRISMA) model was employed, the process of which is shown in Fig. 6.

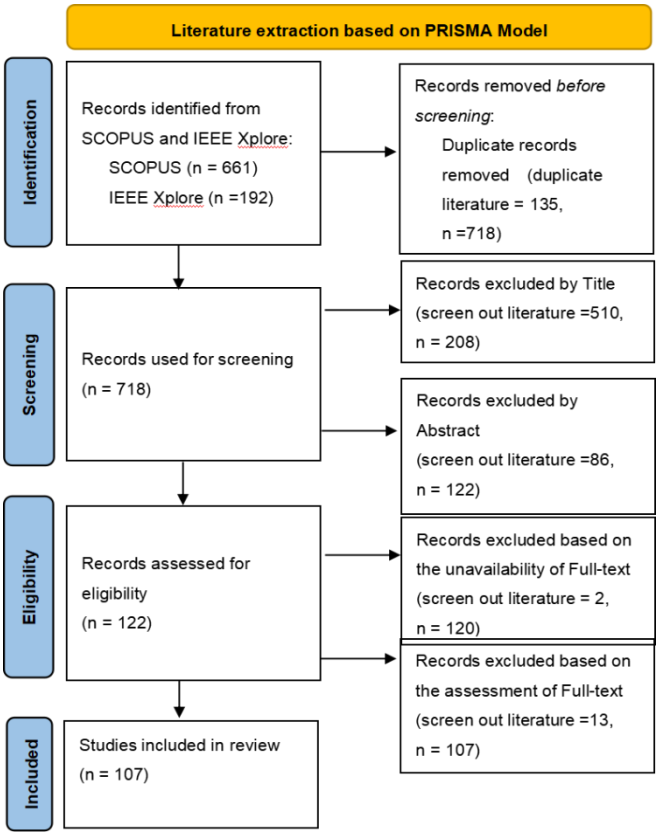


Fig. 6. Literature extraction based on PRISMA model for Study 1.

In the PRISMA Model, there were three crucial stages: identification, screening, and eligibility. At the identification stage, the records were imported into Zotero for identifying duplicate records. During the screening stage, the titles and abstracts of these studies were reviewed. Those that were not highly relevant to the review topic were excluded. At the eligibility stage, based on a full-text review, the studies were assessed. The studies of high quality were included.

After the screening process, 107 studies were selected and included in the subsequent analysis. Conducted over the past 10 years, they focused on factors influencing Chinese consumer adoption intention regarding autonomous vehicles.

D. Process for Extracting Records from Databases for Study 2

Initial searches yielded 890 publications (753 from SCOPUS and 137 from IEEE Xplore). Based on the publication year, research area, document type, and language, the literature was subjected to preliminary screening. We included studies published in English between 2016 and 2025 and within specific document types. Likewise, we selected studies highly relevant to the review topic by applying restrictions on research area. According to these inclusion criteria, 627 studies from SCOPUS and 123 from IEEE Xplore were selected. The process and results are shown in Table II. The PRISMA model was utilized to further screen for appropriate literature, with the process shown in Fig. 7.

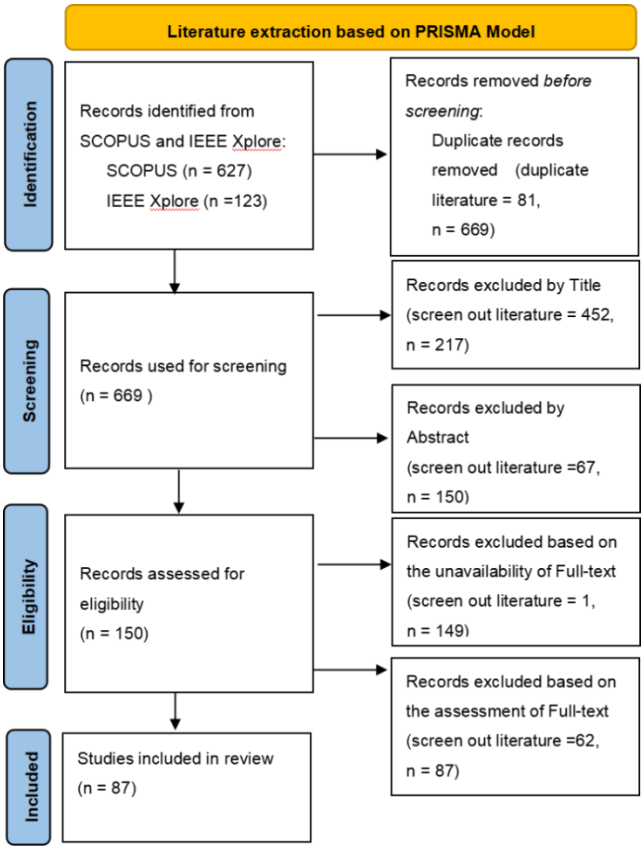


Fig. 7. Literature extraction based on PRISMA model for Study 2.

The studies were selected through three phases: identification, screening, and eligibility. At the identification stage, 750 records were imported into Zotero, and the duplicate records were excluded. During the screening stage, we reviewed the titles and abstracts of these studies and excluded those that were not highly relevant to the review topic. During the eligibility phase, we assessed the quality of these studies based on a full-text review.

After filtering, 87 studies were included in the following analysis. Conducted over a decade, they focused on Chinese consumers' perceptions of autonomous vehicles.

TABLE II. PRELIMINARY SCREENING OF LITERATURE FOR STUDY 2

Filter	SCOPUS	IEEE Xplore
Publication Year	2016-2025 (n = 753)	2016-2025 (n =137)
Research Area	Computer science; Engineering;social sciences; Business, management and accounting; Decision sciences; Economics, econometrics and finance; Psychology Neuroscience (n = 717)	Autonomous Vehicles;Human Drivers;Self-driving;Automated Vehicles(n = 124)
Document Type	Article; Conference paper; book chapter (n =684)	Journals ; Conferences; Early Access Articles; Magazines (n = 123)
Language	English (n = 627)	English (n =123)

IV. BIBLIOMETRIC ANALYSIS: FACTORS INFLUENCING CONSUMER INTENTION TO ADOPT AVS

A. Word Frequency Analysis

Table III shows the frequency and ranking of keywords related to the factors influencing the intention to adopt AVs in the existing literature. According to the results, perception plays a crucial role in influencing consumer adoption intention. The keywords related to perception appear more than 20 times, especially “perceived risk”, which is mentioned 10 times.

The high academic attention devoted to shared autonomous vehicles in China (related concepts were mentioned 13 times, as shown in Table III) demonstrates that consumers have begun to try specific categories of AVs.

The intelligence level of AVs, which is decided by the artificial intelligence with which a car is equipped, proved to be a key factor in consumer adoption intention (“highly autonomous vehicle” appears five times and “artificial intelligence” appears three times). However, AI is not merely a promoting factor in the consumer decision-making mechanism. On the one hand, it can improve consumers’ interest in AVs by making these vehicles smarter and more useful; on the other hand, Chinese consumers are concerned about the effects of anthropomorphism, which may confirm the existence of the uncanny valley theory [24], [26] in the field of autonomous vehicles.

TABLE III. MOST FREQUENT KEYWORDS RELATED TO INFLUENCING FACTORS FOR ADOPTION INTENTION

Words	Occurrences
Shared autonomous vehicle	13
Perceived risk	10
trust	8
Perceived safety	6
Highly autonomous vehicle	5
Perceived value	5
Artificial intelligence	3
Anthropomorphic	2
Covid-19	2
Culture and psychology	2
Mass media	2
Perception	2
Safety	2
Self-efficacy	2
Social influence	2

To increase the intention to adopt AVs among Chinese consumers, internal factors such as building trust (“trust” appeared eight times) and improving self-efficacy (“self-efficacy” was mentioned twice) are very important.

External factors also play an important role. The adoption intention regarding AVs can change due to specific situations, such as the COVID-19 pandemic. Moreover, Chinese culture and psychology, publicity from mass media, and social

influence considerations can influence consumers' behaviors toward AVs.

B. Trend Topic Analysis

According to the changing research trends on topics related to influencing factors shown in Fig. 8, consumer perception emerged as the most extensively researched topic in recent studies examining factors influencing consumer intentions, especially in relation to perceived risk, value, and safety.

Shared autonomous vehicles maintained high popularity and peaked in 2023, but they showed a declining trend in the following year. In contrast, discussions about highly autonomous vehicles have been steadily gaining momentum. This might be a result of recent large-scale pilots of Level 4 autonomous public transportation systems, which have helped change AVs from a concept to reality in consumers' cognition. Many forms of shared autonomous vehicles in China have high intelligence, such as unmanned buses, unmanned taxis and shared school buses. Due to their capacity to reduce driver costs and parking demands, these options are predicted to become the main transportation modes of the future [25].

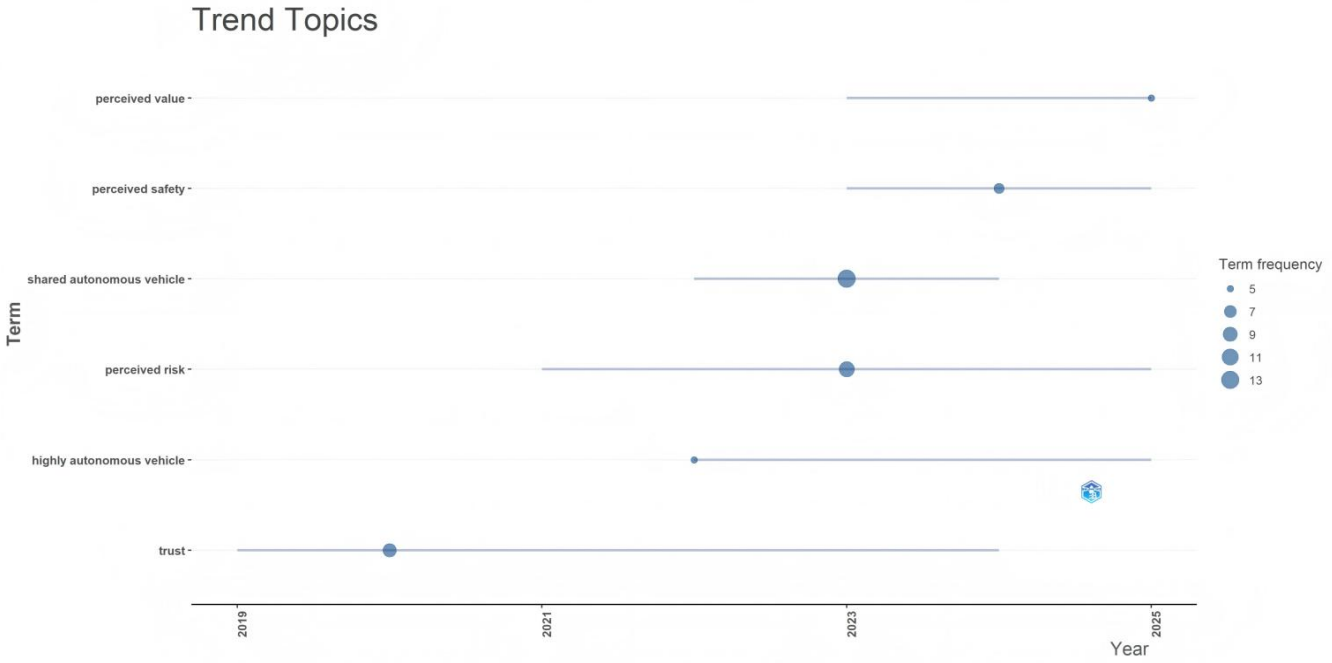


Fig. 8. Trend topics of influencing factors for adoption intention.

V. Bibliometric Analysis: Consumer Perception of AVs

A. Word Cloud Analysis

Fig. 9 shows that for articles related to China published between 2016 and 2025, a large number of consumers' perceptions were related to perceived risk. Words like "perceived usefulness" and "perceived benefit" are also mentioned frequently, according to Fig. 9, but at significantly lower frequency than "perceived risk". The figure also reveals that perceived safety is the category considered most among all the perception dimensions.

The Word Cloud Analysis of keywords also reveals that both internal and external factors are correlated with perceptions. From the perspective of internal factors, attitudes, trust, and demographic characteristics (terms such as 'female,' 'male', and 'adult') contribute the most. From the perspective of external factors, the most important roles are played by social media, as well as economic and social impacts.

Fig. 9 also indicates that current academic research mainly relies on surveys and questionnaires as research tools.

B. Frequency of Words over Time

To better explore how Chinese consumers make decisions based on perceptions, the synonym tool in the Bibliometrix was used to categorize keywords by theme. Frequency analysis over time was conducted related to consumer perceptions in the Chinese autonomous vehicle market, as shown in Fig. 10. This study grouped words about negative factors (like "perceived risk", "perceived cost", and "perceived loss") as "perceived loss", while positive words (like "perceived benefit", "perceived usefulness", and "perceived gain") were grouped as "perceived gain".

This analysis demonstrates a clear relationship between perceived gain and perceived loss. Although the frequencies of these two concepts were close to each other in 2018, they indicated different trends thereafter. The cumulative occurrence of "perceived gain" remained stable from 2020, while that of "perceived loss" obviously increased during the same period, surpassing that of "perceived gain" in 2022. Perceived safety, as the most-frequently considered dimension of perceptions, also revealed a significant upward trend since 2020, surpassing "perceived gain" in 2024.



Fig. 9. Word cloud of keywords related to consumer perception.

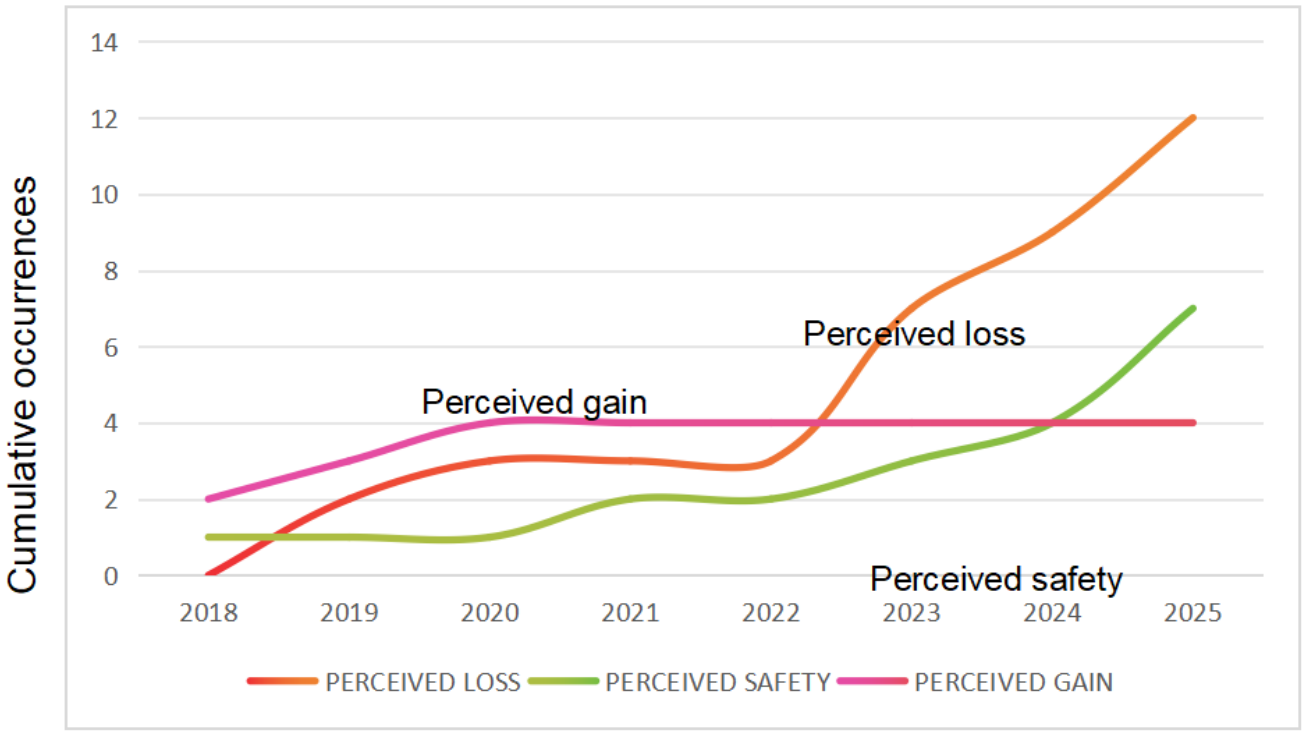


Fig. 10. Keywords' frequency over time related to consumer perception.

C. *Keyword Co-occurrence Network*

Fig. 11 shows a co-occurrence network related to Chinese consumers' perceptions of the autonomous vehicle market. On the one hand, it demonstrates strong relationships among perceived loss, perceived safety, traffic accidents, and demographic characteristics. The results suggest that Chinese residents, including different gender and age groups (female, male, young adults, middle-aged adults, and aged people), are more concerned about traffic risks caused by AVs than the safety they bring. This result is also an evidence proving that Chinese consumers are more sensitive to losses than to gains.

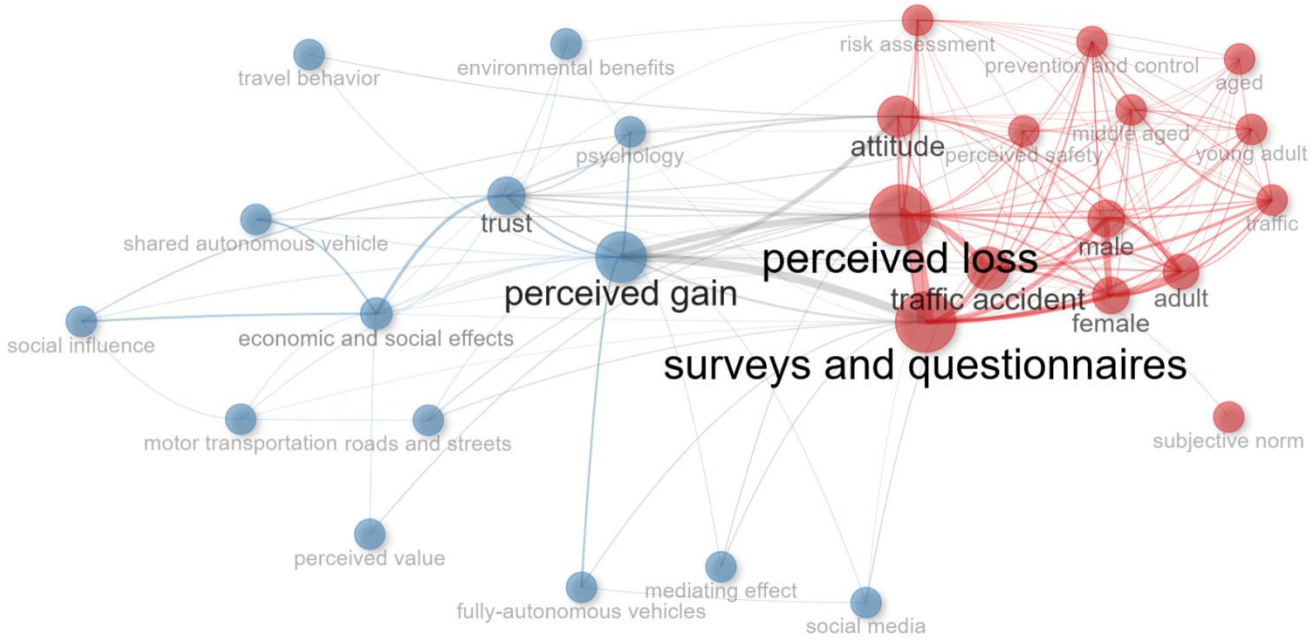


Fig. 11. Co-occurrence network related to consumer perception.

Furthermore, Fig. 11 shows that perceived gain has deep connections with economic and social effects, environmental benefits, and social influence in China. The term "share autonomous vehicle" is also in this category. This means that Chinese consumers develop more positive perceptions if they believe that, first, AVs can benefit society and the environment and, second, that taking shared autonomous vehicles is a reasonable and attractive option.

VI. Conclusion and Implications

In conclusion, the study has resolved all the research questions and demonstrates several points.

Through bibliometric analysis, numerous factors were found to influence Chinese consumers' adoption intention regarding AVs, including perception, intelligence level, trust, and certain external factors. Consumer perception seems the most important factor. The subsequent bibliometric analysis indicated the perception mechanism of Chinese consumers. The results showed that they judged perceived loss to be more serious than perceived gain. These consumers also showed significant concerns about perceived safety and the influence of AVs on both society and the environment.

Theoretically, this study systematically integrates literature on consumer adoption of AVs in the specific area of China, thus bridging a knowledge gap. In practical terms, this study can improve how governments and enterprises understand the behaviors and decision-making mechanisms of Chinese consumers in the autonomous vehicle market, thereby accelerating the diffusion of autonomous vehicle technology.

By reviewing the existing literature, we have identified the main trends and significant research gaps in the autonomous vehicle field in China.

First, there is a strong focus on shared autonomous vehicles. As one of the crucial application scenarios of AVs, it has received significant academic attention. However, based on the diffusion of innovation theory, purchasing and owning private AVs are also vital factors influencing the development of the Chinese AV market. Moreover, there is a need to expand the use cases for shared autonomous vehicles, not limited to school buses or robo-taxis.

Second, most empirical methods and tools for measuring consumer perceptions remain surveys and questionnaires. However, overreliance on this method could introduce biases, such as self-report bias, intention-behavior gap, and social desirability bias. Higher quality scales need to be developed, and additional empirical methods need to be applied in this field.

This study proposes that future research should expand the application scenarios of AVs, especially for private AVs and use cases for shared AVs, such as applications in rural areas. Additionally, other empirical methods, such as longitudinal studies and mixed methods, should also be used in this area to reduce bias.

Due to the limitations caused by the brief development duration of the autonomous vehicle industry and the geographical scope of this research, the number of documents included in the two bibliometric analyses were relatively small. Future research will aim to add other relevant documents, such as government reports, to provide stronger support for the conclusions. Additionally, empirical methods will be used to strengthen the conclusions in the future.

Acknowledgment

The authors would like to thank anonymous reviewers for their valuable comments and suggestions references.

REFERENCES

[1] Talavera, E., Díaz-Álvarez, A., Naranjo, J. E., & Olaverri-Monreal, C. (2021). Autonomous vehicles technological trends. Electronics, 10(10), 1207.

[2] Ilková, V., & Ilka, A. (2017, June). Legal aspects of autonomous vehicles−An overview. In 2017 21st international conference on process control (PC) (pp. 428-433). IEEE.

[3] Mobilus(2018).Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles.SAE International.

[4] Stoma, M., Dudziak, A., Caban, J., & Droździel, P. (2021). The future of autonomous vehicles in the opinion of automotive market users. Energies, 14(16), 4777.

[5] Mezghani, Zhao (2024, October). How autonomous vehicles can be integrated with public transport systems for urban mobility. World Economic Forum. Retrieved from https://www.weforum.org/stories/2024/10/how-will-autonomous-vehicles-shape-urban-mobility on October 16, 2024.

[6] Rajpal. (2024, October). Mainland China autonomous vehicle development on a different track. S&P Global Mobility. https://www.spglobal.com/mobility/en/research-analysis/china-autonomous-vehicles-development.html.

[7] Huang, K., Kockelman, K., & Gurumurthy, K. M. (2023). Innovations impacting the future of transportation: an overview of connected, automated, shared, and electric technologies. Transportation Letters, 15(6), 490-509.

[8] Pizzuto, L., Thomas, C., Wang, A., & Wu, T. (2019, January). How China will help fuel the revolution in autonomous vehicles. McKinsey & Company.

[9] Li, Y. (2024, July 30). Carrot run: Why China is launching large-scale driverless taxi trials. BBC News. https://www.bbc.com/zhongwen/trad/chinese-news-69226660.

[10] Counterpoint Research. (2024). Counterpoint Research's latest global autonomous passenger vehicle tracker and forecast [Report]. https://www.counterpointresearch.com/insights/adas-market-china-shifting-gears-level3/.

[11] Xie, Y., & Xu, W. (2023, November 12). China's unmanned delivery technology emerges during "Double 11" shopping festival. Xinhua News. http://www.news.cn/fortune/2023-11/12/c_1129971132.htm.

[12] Renub Research. (2024). China autonomous vehicles market report by level of driving (L1, L2, L3, L4, L5), hardware (passive components, embedded modem, ultrasonic sensors, odometry sensors, other electronics & architecture, actuators, HMI hardware, mapping hardware, embedded controls hardware, V2X hardware, cameras, radar & lidar), software (HMI software, data security software, mapping software, embedded controls software & V2X software), vehicle type, application, propulsion, and companies analysis 2025 - 2033 [Report]. https://www.renub.com/china-autonomous-vehicles-market-p.php.

[13] Ipsos. (2018). Most global consumers are intrigued by the idea of self-driving cars [Report]. https://www.ipsos.com/en/most-global-consumers-are-intrigued-idea-self-driving-cars.

[14] Ipsos. (2022). Future of mobility: Autonomous driving and the impact on our life [Report]. https://www.ipsos.com/en-my/future-mobility-autonomous-driving-and-impact-our-life.

[15] Siddharta, A. (2022, October 5). Share of respondents in Malaysia who thought it likely that self-driving cars would be a common sight in 2020. Statista. https://www.statista.com/statistics/1090763/malaysia-likelihood-self-driving-cars-being-the-norm/.

[16] Topolšek, D., Babić, D., Babić, D., & Cvahte Ojsteršek, T. (2020). Factors influencing the purchase intention of autonomous cars. Sustainability, 12(24), 10303.

[17] Jing, P., Xu, G., Chen, Y., Shi, Y., & Zhan, F. (2020). The determinants behind the acceptance of autonomous vehicles: A systematic review. Sustainability, 12(5), 1719.

[18] Islam, M. R., Abdel-Aty, M., Lee, J., Wu, Y., Yue, L., & Cai, Q. (2022). Perception of people from educational institution regarding autonomous vehicles. Transportation research interdisciplinary perspectives, 14, 100620.

[19] Chikaraishi, M., Khan, D., Yasuda, B., & Fujiwara, A. (2020). Risk perception and social acceptability of autonomous vehicles: A case study in Hiroshima, Japan. Transport Policy, 98, 105-115.

[20] Cai, Z., & Xiong, A. (2023). Understand Users' Privacy Perception and Decision of {V2X} Communication in Connected Autonomous Vehicles. In 32nd USENIX Security Symposium (USENIX Security 23) (pp. 2975-2992).

[21] He, Q. (2024). A review of research on consumer perceived value. Accounting and Corporate Management, 6(3), 7-15.

[22] Framarz, B., Mohamed, A., Krish, K., & Henry, W. (2016). A Meta-Analytical Approach toward development of a Comprehensive Measurement Scale for Consumers' Perceived Risks from Innovative Offerings. Advances in Management, 9(7), 1.

[23] Lee, H., Im, E., Oh, M., & Gim, G. (2023, December). A Study on the Intention to Utilize Overseas Developers Through Offshoring−Using the Value-Acceptance Model (VAM). In IEEE/ACIS International Conference on Big Data, Cloud Computing, and Data Science Engineering (pp. 55-66). Cham: Springer Nature Switzerland.

[24] Aria, M., & Cuccurullo, C. (2017). bibliometrix: An R-tool for comprehensive science mapping analysis. Journal of Informetrics, 11(4), 959-975.

[25] Trancoso, B. I. F. F. D. (2024). Exploring the uncanny valley effect: how visual perception related autistic traits shape human-robot interaction (Doctoral dissertation).

[26] Huang, K., Kockelman, K., & Gurumurthy, K. M. (2023). Innovations impacting the future of transportation: an overview of connected, automated, shared, and electric technologies. Transportation Letters, 15(6), 490-509.

RA-ACS_net Network: A Quantum Optical Reconstruction Method for Ultra-high Resolution Bioimaging

Lin SHANG*
School of Electronics and Information Engineering, Wuhan Technical College of Communications, Wuhan 430065, Hubei, China

***Abstract*—Ultra-high resolution bioimaging based on quantum optics offers high sensitivity at relatively low cost, yet conventional reconstruction algorithms face challenges of excessive sampling time, long computation, and artifacts that limit imaging quality. To overcome these issues, this study proposes a novel quantum optical bioimaging reconstruction method termed RA-ACS_net, which integrates a ripple algorithm with a hybrid attention mechanism network. The ripple algorithm provides global optimization for network parameter adjustment, while the attention mechanism enhances feature extraction and information fusion. Furthermore, a differentiated loss function (ALoss) is designed to preserve fine structural details and improve visual fidelity compared with conventional MSE loss. A large-scale dataset of quantum optics-based bioimages is employed for training and validation. Experimental results demonstrate that RA-ACS_net achieves superior reconstruction performance, with significantly higher PSNR and SSIM across both low and high sampling ratios, when compared to iterative algorithms (TVAL3) and existing deep learning models (DR2-Net, DPA-Net). The proposed approach exhibits robustness under sparse data conditions, reduces blocking artifacts, and accelerates convergence, thereby addressing critical limitations of current methods. This study highlights the potential of combining quantum optics with advanced deep learning optimization strategies to establish a practical and efficient framework for ultra-high resolution bioimaging.**

Keyword*—*Ultra-high resolution bioimaging; quantum optics; computer vision; ripple algorithm; attention mechanism

I. Introduction

Bioimaging techniques are the cornerstone of modern biological and medical research, enabling us to observe and understand the structure and function of living systems at the molecular, cellular and tissue levels [1]. However, traditional imaging techniques, such as fluorescence microscopy and electron microscopy, are limited by diffraction limits and imaging speeds, making it difficult to meet the demand for high-resolution and highly dynamic imaging of biological systems [2]. In recent years, the rapid development of quantum optics and computer vision technology has provided the possibility to break through these limitations [3]. Quantum optics, especially single-photon detection technology and quantum entanglement, provides a means for bioimaging to go beyond the traditional diffraction limit; while computer vision technology, on the other hand, shows great potential in processing and analysing massive biological image data [4]. Therefore, it is very meaningful to study the integration method between ultra-high resolution bio-imaging, quantum optics and computer vision techniques, especially the use of deep learning techniques to generate models for ultra-high resolution bio-imaging image reconstruction [5].

With the application and development of quantum optics and computer vision technology in the field of bio-imaging, the research on quantum optics and computer vision technology for ultra-high resolution bio-imaging has also attracted the attention of experts and scholars in the field [6]. Currently, the research on the application and integration of quantum optics and computer vision technology in the field of bio-imaging mainly includes the application of quantum optics in bio-imaging, bio-imaging reconstruction based on machine vision, and bio-imaging image feature extraction and recognition [7]. This study focuses on the ultra-high resolution bio-imaging reconstruction problem by fusing quantum optics, and analyses a large number of literature. Currently, there are three main methods for sampling and reconstruction of quantum optics bio-imaging systems: 1) Under-sampling using a random measurement matrix or an orthogonal fixed matrix first, and then using iterative algorithms to reconstruct the images. Chong and Pramanik [8] use Gaussian measurement matrices to undersample the bioimaging, and orthogonal matching tracking method is used to reconstruct the image; Wanas et al, [9] use the full-variance generalised Lagrangian alternating direction algorithm as an iterative algorithm to reconstruct the bioimaging; 2) using Gaussian random measurement matrices or orthogonal fixation matrices for under-sampling, but using the deep learning-based reconstruction network to reconstruct the image. Yao et al. [10] proposed DR2-Net based on ReconNet, which solves the problem of degradation of reconstructed image accuracy due to the excessive depth of the network; Sun et al. [11] proposed a dual-path attentional network DPA-Net for compressed perceptual image reconstruction, which consists of a structural path, a texture path, and a texture-attentional module; and 3) using a deep learning network that integrates sampling and reconstruction. A sampling and reconstruction integrated residual codec network SRIED_Net was proposed in the literature [12] and used in a quantum optical bio-imaging system, where the results showed that the network reconstructs quickly and obtains better reconstruction performance. The current network reconstructs the image resulting in blocking of the reconstructed image, which reduces the reconstruction accuracy of quantum optics-based high-resolution bioimaging.

Aiming at the problems of the current high-resolution bio-imaging reconstruction network based on quantum optics, this study proposes a ripple algorithm to optimise the attention mechanism network for quantum optics high-resolution bio-imaging reconstruction method. Firstly, the problem of quantum optics application for ultra-high resolution bio-imaging, secondly, combining the ripple algorithm [13] and attention mechanism [14], the RA-ACS_net reconstruction network is proposed, and finally, the effectiveness and efficiency of the RA-ACS_net reconstruction network are verified by using high-resolution bio-imaging data based on quantum optics.

The main novelty of this study lies in the integration of the ripple algorithm with a hybrid attention mechanism network to address the limitations of existing quantum optical bioimaging reconstruction methods. Unlike traditional iterative algorithms and standard deep learning frameworks that often suffer from long reconstruction times, blocking artifacts, and limited robustness, the proposed RA-ACS_net model introduces an optimized parameter adjustment process guided by ripple-based global search. In addition, a novel differentiated loss function, ALoss, is designed to preserve structural details and enhance perceptual quality beyond the performance of conventional MSE. Experimental results on quantum optical bioimaging datasets demonstrate that the RA-ACS_net network significantly outperforms state-of-the-art models under both low and high sampling ratios, achieving higher PSNR and SSIM values. This methodological advancement not only improves reconstruction accuracy and robustness but also offers a practical pathway for bridging quantum optics and advanced computer vision in ultra-high resolution bioimaging.

The full study is structured as follows: firstly, the application of quantum optics in bio-imaging is analysed in detail in Section II, including single photon detection technology, quantum entanglement imaging and its practical application in biological systems. The design and realisation of the quantum optical bio-imaging system is presented, focusing on the core devices of the system and their working principles. In Section III, the RA-ACS_net reconfiguration network is proposed, and the reconfiguration network is optimised by combining the ripple algorithm and the attention mechanism. The bio-imaging reconstruction approach is presented in Section IV. Section V verifies the performance enhancement of the method proposed in this study in quantum optical bio-imaging through experimental comparison, and finally, Section VI summarises the research results of this study and provides an outlook on the future research direction.

II. Quantum Optics for Ultra-High Resolution Bioimaging Analysis

A. Quantum Optics in Biological Imaging

1) Single-photon detection technology: Single-photon detection technology takes advantage of the quantum properties of photons and is able to detect a single photon as a unit, thus achieving high-precision imaging under very low light intensity conditions [15], as shown in Fig. 1. In biological imaging, this technique is particularly suitable for fluorescently labelled single-molecule imaging, which can obtain high-resolution structural information while maintaining cellular activity.

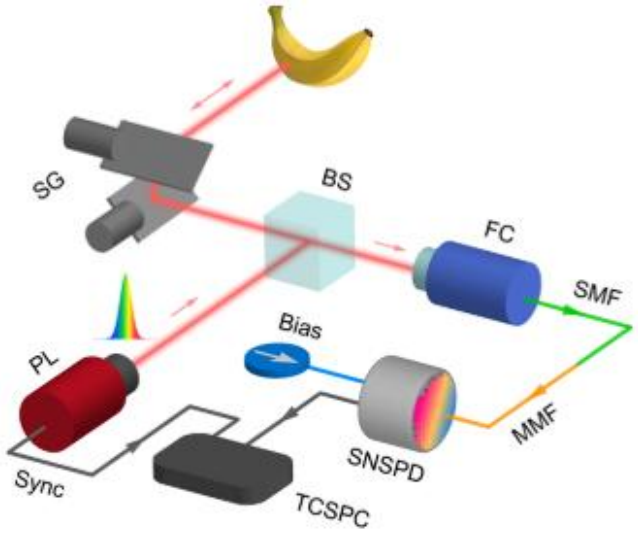


Fig. 1. Single-photon detection technology.

2) Quantum entanglement and imaging: Quantum entanglement [16] is a fundamental concept in quantum optics, where two or more particles are interdependent on each other in their quantum states, and even if there is a great distance between them, measurements on one of the particles will instantly affect the state of the other, and the schematic diagram of the principle of quantum entanglement is shown in Fig. 2.

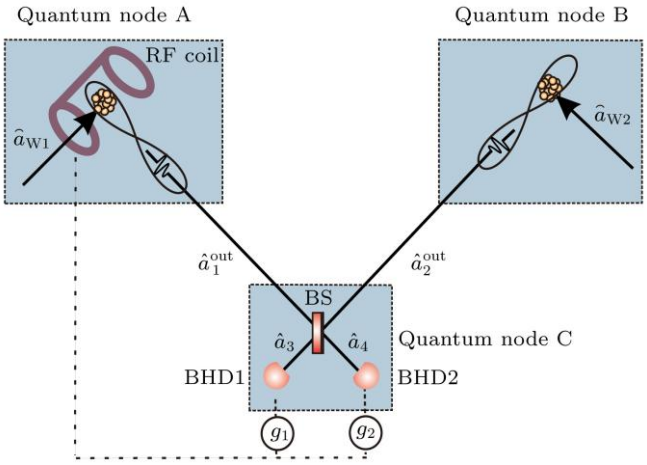


Fig. 2. Quantum entanglement.

Using the properties of quantum entanglement, a super-resolution quantum imaging system can be designed (Fig. 3) to achieve high-resolution imaging beyond the diffraction limit.

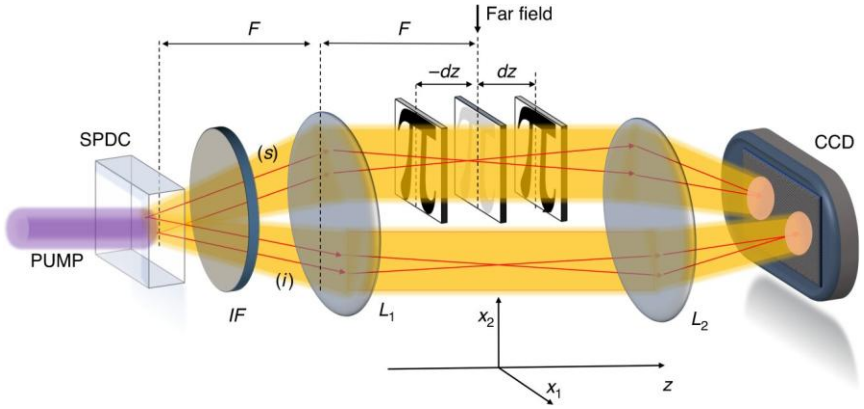


Fig. 3. Quantum imaging system.

3) Quantum optics in bioimaging: Quantum optical techniques show great potential in the field of bio-imaging due to their unique quantum coherence and entanglement effects. These techniques can significantly improve the resolution and sensitivity of imaging beyond the limits of conventional optical physics. Quantum imaging techniques using entangled photon pairs and single-photon detection can enable the observation of virus-sized cellular tissue structures as well as nanoscale defects in insulator materials, with specific applications analysed, as shown in Fig. 4.

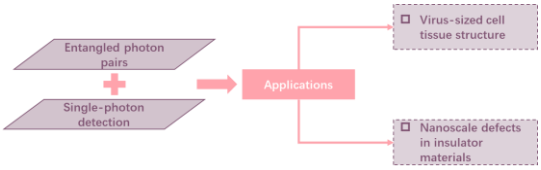


Fig. 4. Application of quantum optics in bio-imaging.

B. Quantum Optical Bio-Imaging System

Quantum optical bio-imaging systems utilise quantum coherence and entanglement effects to improve the resolution and sensitivity of imaging beyond the limits of conventional optical physics [17]. The research object of this study is the quantum optical single-pixel imaging system, which is shown in Fig. 5.

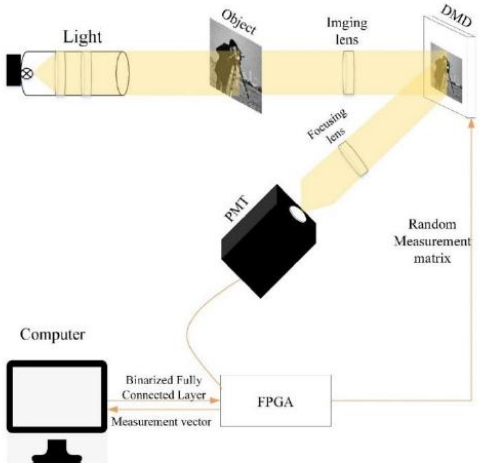


Fig. 5. Quantum optical single-pixel imaging system.

This study builds a quantum optical bio-imaging system whose core devices mainly contain a parallel single-photon source, a digital micromirror device DMD, and a single-photon detector PMT.

1) Parallel single photon source: In order to reduce the impact of light source scattering on the imaging results, this study designs a parallel single-photon source that can output a very weak parallel light, with a specific structure principle, as shown in Fig. 6.

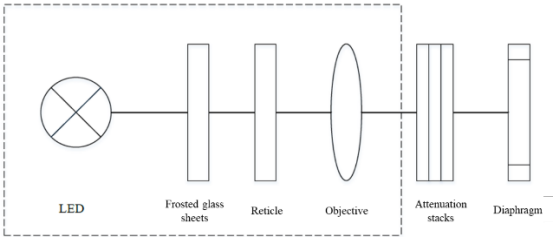


Fig. 6. Structure of parallel single-photon source.

2) Digital micromirror device DMD: Photonic single-pixel imaging schemes mainly implement spatial light modulation technology through spatial light modulators [18]. The electrically addressed spatial light modulator DMD used in this study has the advantages of high brightness, high reliability, simplified working circuit and high contrast, and its specific structure is shown in Fig. 7. The parameter settings of the DMD development board used in the experimental system of this study are shown in Table I.

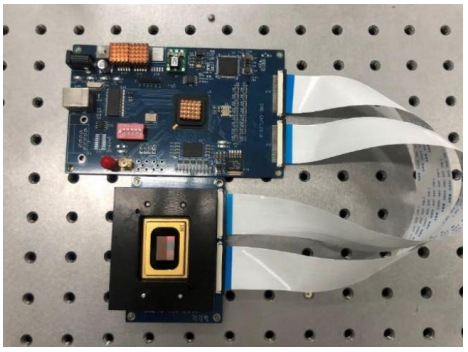


Fig. 7. Structure of DMD development board.

TABLE I DMD OPERATING PARAMETERS

Sports event	Setting parameters
resolution (of a photo)	1024×768
Micromirror size	13.68μm×13.68μm
Flip angle	±12°
frame rate	3000Hz
degree of contrast	2000:1
operating band	350nm~2700nm
synchronisation	Internal synchronisation, external synchronisation
random access memory (RAM)	DDR2

3) Single-photon detector PMT: The photon-counting single-pixel imaging system utilises a detector with a single-photon response to achieve the quantification of modulated light intensity. In this study, PMT is chosen as a single-photon detector, which is mainly composed of a photocathode, a collection anode and an electron multiplier pole, and its internal structure is shown in Fig. 8:

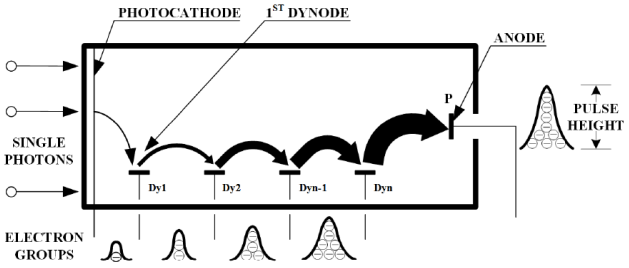


Fig. 8. PMT schematic.

C. Quantum Optical Bioimaging Reconstruction

In quantum optical bio-imaging systems, the traditional iterative reconstruction algorithms, although capable of accurately reconstructing the target image, have high time complexity, especially for large resolution images, where the reconstruction time is exponentially multiplied. In order to improve the efficiency of image reconstruction, ReconNet, as a neural network for reconstructing compressed perceptual images, was introduced into the quantum optical bioimaging reconstruction problem [19].

The structure of the ReconNet network is shown in Fig. 9. ReconNet consists of an upsampling module and an image enhancement module. The upsampling module contains one fully connected layer, and the image enhancement module contains six convolutional layers, each of which is post-positioned with a ReLU activation function to perform a nonlinear transformation.

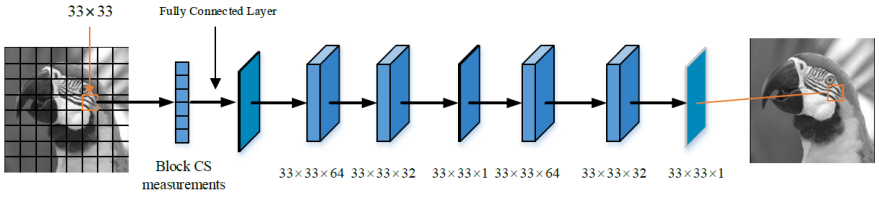


Fig. 9. ReconNet structure.

III. RA-ACS_NET RECONFIGURATION NETWORK

In order to improve the accuracy and efficiency of ultra-high resolution bio-imaging reconstruction based on quantum optics, this study uses the ripple algorithm to optimise the attention mechanism network and construct the RA-ACS_net reconstruction network.

A. Network of Attention Mechanisms

In order to reduce the computational burden of the reconstruction network and make full use of the image summary to give important information, this study adopts the attention mechanism [20] to construct the image reconstruction network and improve the image reconstruction effect. Attention mechanism network is divided into Hard Attention mechanism (Hard Attention) and Soft Attention mechanism (Soft Attention). According to the classification of attention mechanism structure and function, the attention mechanism network can be divided into Channel Attention Mechanism, Spatial Attention Mechanism and Hybrid Attention Mechanism, and the specific structure schematic is shown in Fig. 10.

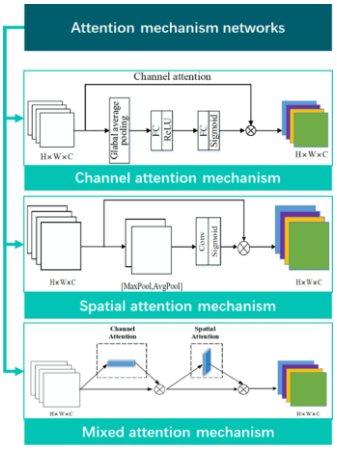


Fig. 10. Classification of attention mechanisms.

Inspired by the hybrid attention mechanism, this study adopts an attention mechanism network based on residual neural network, i.e., ACS_net, and the specific network structure is shown in Fig. 11. As can be seen from Fig. 11, ACS_net contains a picking sub-network and a deep reconstruction sub-network.

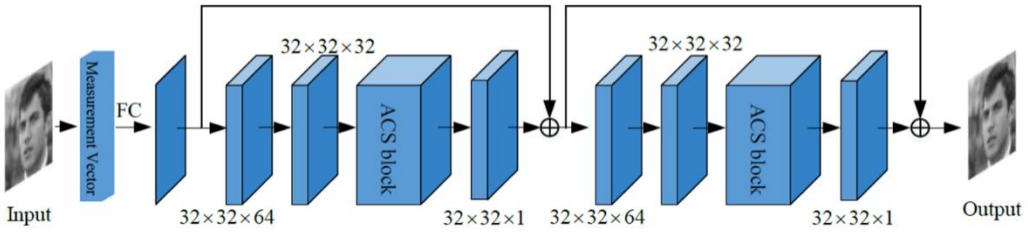


Fig. 11. ACS_net structure.

In the sampling network part, considering the screening of the input image information to ensure the completeness of the input information, ACS_net adopts the full connection layer to sample the original image to ensure the completeness.

In the deep reconstruction subnetwork part, ACS_net employs two residual blocks [21] to achieve the combination of lower-order features with higher-order features through jump connections. Each residual block contains a hybrid attention mechanism module (ACS_block) [22] and three convolutional layers. The structure of ACS_block is shown in Fig. 12, in which two convolutional layers are front-loaded, and then Batch Normalizaition is performed on these two convolutional layers, followed by a nonlinear transformation of the input feature maps through the activation function ReLU; finally, the input features are processed using the convolutional layers and the image reconstruction results are obtained.

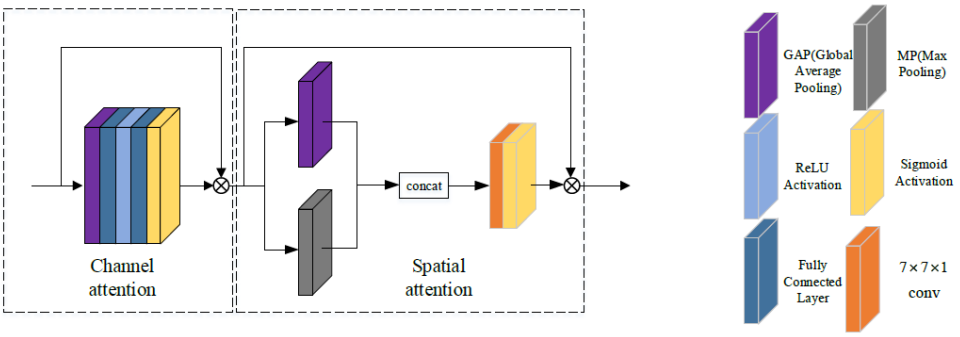


Fig. 12. Structure of the attention mechanism module.

In order to improve the overall performance of the model, this study designs a new loss function, namely ALoss (Attention Loss), which is calculated as follows, Eq. (1):

$$ALoss = \alpha \times \frac{1}{N}\sum_{i=1}^{N}\left\|x_i' - x_i\right\|^2 + \beta \times \frac{1}{N}\sum_{j=1}^{N}\left\|x_j'' - x_j\right\|^2 \quad (1)$$

where, x_i and x_j are the pixel values of the important and non-important regions of the original image block, x_i' and x_j'' are the pixel values of the important and non-important regions of the image recovered by the neural network, α and β are the scaling coefficients of the loss function, and N is the number of pixel points in the focal and non-focal regions of the image.

B. Ripple Algorithm

Ripple Algorithm (RA) is an optimization algorithm that simulates the phenomenon of ripple propagation in nature [13]. The RA algorithm has two features:

- The centroid adopts the ripple region search based on the current fitness and the global optimal fitness, which has stronger global search capability.
- The centroid searches relatively independently with the current optimal solution, and converges to the global optimal solution in a passive manner.

According to the characteristics of the ripple algorithm, this study performs a multi-layer ripple random point taking search with each centre point and calculates the random point adaptation value. The RA algorithm mainly includes two aspects of the search capability, i.e., exploration and exploitation, and the specific search schematic is shown in Fig. 13.

The multi-layer ripples are calculated by taking random points, as follows in Eq. (2):

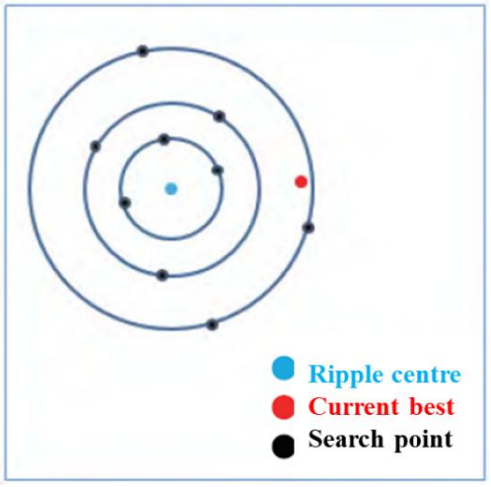
$$X_i^{km} = X_i + r \times R_i^k \quad (2)$$

where, X_i^{km} denotes the mth random search point in the kth layer of the ith centroid, X_i is the ith centroid, R_i^k is the radius of the kth layer of the ith centroid, and r is an n-dimensional random unit vector.

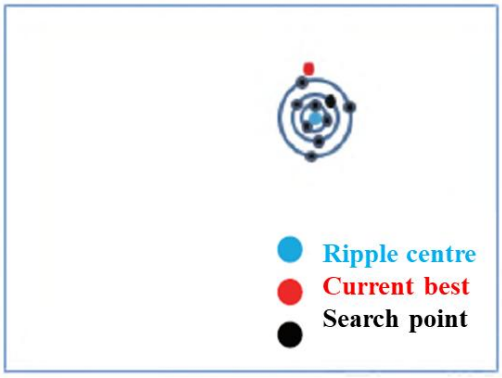
The ripple radius of each centre point is based on the gap between the current fitness and the global optimal fitness, the larger the gap, the larger the ripple radius and the wider the search area, and the smaller the gap, the smaller the ripple radius and the smaller the search area. The equation for calculating the ripple radius is as follows in Eq. (3):

$$R_i^1 = \log_\alpha \left(\frac{\sigma_i}{\sigma_{\max}} \right) \times \beta \tag{3}$$

where, α is the base of the logarithmic function and $\alpha \in (0,1)$; $\sigma_{\max}$ denotes the global maximum fitness value.



(a) Exploration



(b) Development

Fig. 13. RA algorithm search.

Comparing each centroid with its corresponding random point in the ripple layer, the point with the best fitness value is retained to form a new centroid cluster. Each centroid searches for the optimal point relatively independently rather than actively approaching the current optimal point, and the centroid cluster eventually converges to the optimal point as the iteration proceeds with the shrinkage function decreasing. The centre point update is calculated as follows in Eq. (4):

$$X_i^{t+1} = opt\left(X_i^t, X_i^{11}, \cdots, X_i^{km}, \cdots, X_i^{3M} \right) \tag{4}$$

where, X_i^{t+1} is the ith centroid of the centroid cluster in generation t+1, X_i^t is the ith centroid of the centroid cluster in generation t, X_i^{km} is the mth search point in layer k corresponding to the ith centroid, and M is the number of randomly taken points for each layer of ripples.

C. RA-ACS_Net Network Model

In order to improve the efficiency of ACS_net network reconfiguration, this study adopts the RA algorithm to optimise the ACS_net network parameters. The RA algorithm takes the ACS_net network parameters as the optimisation variables, MSE as the optimisation fitness function, and adopts the iterative process of the RA algorithm as the optimisation strategy, and the specific model construction is shown in Fig. 14.

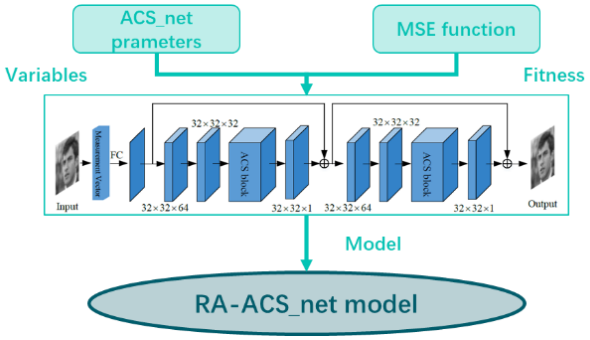


Fig. 14. RA-ACS_net network model construction diagram.

IV. BIO-IMAGING RECONSTRUCTION APPROACH

In order to improve the efficiency and accuracy of the fusion quantum optics ultra-high resolution bio-imaging reconstruction method [23], this study proposes a fusion quantum optics ultra-high resolution bio-imaging reconstruction method based on RA-ACS_net. The method combines quantum optics, constructs an ultra-high resolution bio-imaging system, optimises the ACS_net network parameters using the ripple algorithm, and constructs a bio-imaging reconstruction method based on the RA-ACS_net network model. The specific method flow is shown in Fig. 15.

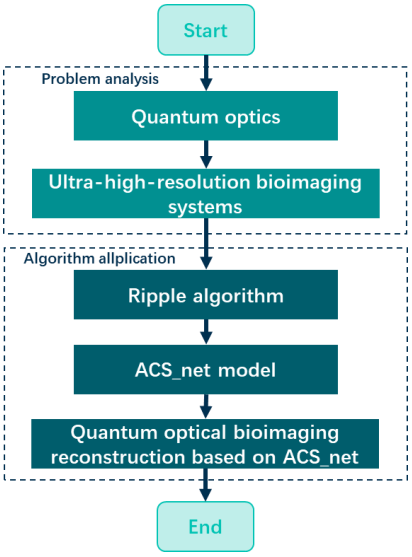


Fig. 15. Application of RA-ACS_net model in quantum optical bio-imaging.

V. Analysis of Results

In order to verify the efficiency of the RA-ACS_net reconstruction network proposed in this study, this study collects 19872 quantum optics-based bio-imaging images as the training set, and 128 bio-imaging images as the dataset for the training set to verify the performance of the network. For comparative analysis, this study firstly investigates the performance of different attention mechanism networks based on the RA algorithm in bio-imaging image reconstruction algorithms, secondly investigates the effect of different loss functions on reconstruction networks, and finally verifies the performance of the attention mechanism in quantum optics-based bio-imaging.

A. Comparative Experiments on Networks with Different Attention Mechanisms

In order to evaluate the effect of the attention mechanism on the reconstruction network models proposed in this study, controlled experiments are designed in this section to compare the reconstruction performance of the four reconstruction network models. The comparison algorithms used in the different attention mechanism network comparison experiments are shown in Table II. The results of the experiments of different attention mechanism network modules are shown in Fig. 16.

TABLE II Parameter Settings of the Algorithm

Arithmetic	Algorithm setup
RA-CS_net	Removal of the Attention Mechanism module and optimisation of the CS_net parameter using the RA algorithm
RA-ACS_net_V1	Contains only the Channel Attention Mechanism module, which uses the RA algorithm to optimise the ACS_net_V1 parameter
RA-ACS_net_V2	Contains only the Spatial Attention Mechanism module, which uses the RA algorithm to optimise the ACS_net_V2 parameter
RA-ACS_net	Contains a hybrid attention mechanism module with RA algorithm to optimise ACS_net parameters

Fig. 16 shows the performance comparison of the four models at different Measuring Ratio (MR). It can be clearly seen that as the Measuring Ratio increases, the PSNR of all the models gradually increases. This is in line with the principle of Compressed Sensing (CS): the higher the Measuring Ratio, the more raw data we acquire, the better the quality of the reconstructed image, and the PSNR rises.

The RA-CS_net model performs the worst over the entire range of measurement ratios, especially at low measurement ratios (e.g., 0.1), where its PSNR is only about 25 dB, which is significantly lower than the other models. Even though the PSNR of this model increases as the measurement ratio increases, it is smaller and eventually the PSNR is still below 30 dB at a measurement ratio of 0.5. This indicates that the RA-CS_net model is weak in reconstructing the image with limited information, probably because the architecture of the model loses a lot of information when dealing with high compression rates.

By comparing RA-ACS_net_V1 with RA-ACS_net_V2, we find that both versions of the improved model outperform RA-CS_net over the entire range of measurement ratios, especially at a measurement ratio of 0.1, where the PSNR already reaches about 28 dB. As the measurement ratio increases, their PSNR improves rapidly, and finally at a measurement ratio of 0.5, both of them reach a PSNR of about 32 dB, which significantly outperforms that of RA-CS_net, indicating that the ACS (Adaptive Compressed Sensing) architectures have a significant improvement in the image reconstruction task, and they can better recover the original image, especially at medium to high measurement ratios.

RA-ACS_net: the model has the most impressive performance, especially at higher measurement ratios (e.g. 0.4-0.5), where it achieves a PSNR of more than 34 dB and close to 35 dB. This suggests that RA-ACS_net is able to utilise more measurements more efficiently, further improving the quality of the reconstruction.

The different magnitudes of PSNR enhancement with measurement ratio for different models are also reflected in Fig. 16. RA-CS_net has a relatively flat enhancement, indicating that it is not efficient in utilising the additional data. RA-ACS_net_V1 and RA-ACS_net_V2 have faster enhancement rates, especially at measurement ratios from 0.1 to 0.3, and their PSNRs are rapidly enhanced, which indicates that these improved models have a greater increase in data utilisation.RA-ACS_net boosts the most, and its PSNR continues to increase at higher measurement ratios (0.3 to 0.5), indicating that it performs best in high measurement ratio scenarios.

At low measurement ratios (0.1), the differences between the models are more obvious, especially RA-CS_net is much lower than the other three models. This indicates that the traditional RA-CS architecture has been difficult to recover images effectively when the amount of data is extremely limited, whereas the ACS-based model is able to extract more useful information from a small amount of data by means of self-adaptation, which significantly improves the reconstruction quality.

When the measurement ratio reaches 0.5, RA-ACS_net performs the best, with a PSNR close to 35 dB, while RA-CS_net still stays at less than 30 dB.This suggests that, although the reconstruction quality of all the models improves at high measurement ratios, the architecture of RA-ACS_net is able to make better use of the additional measurements and show better reconstruction results. This may be due to the stronger feature extraction and information fusion capabilities of the model.

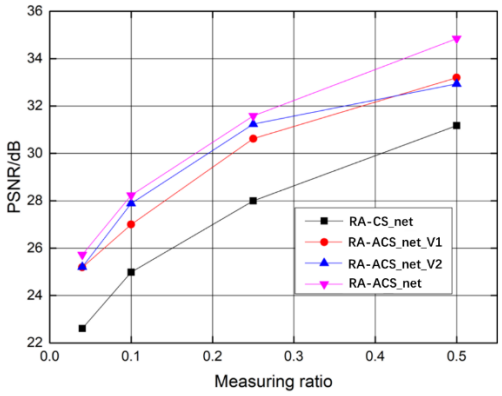


Fig. 16. Peak signal-to-noise ratio of reconstructed networks with different attention mechanism modules.

B. Comparison Experiment of Different Loss Functions

In order to evaluate the impact of the differentiated loss function ALoss designed in this study on the performance of the reconstructed network, a set of controlled experiments is designed in this section for validation, i.e., the comparison experiments between RA-ACS_net+MSE and RA-ACS_net+ALoss. The specific results are shown in Table III and Table IV.

From Table III, it can be seen that in terms of PSNR, the peak signal-to-noise ratio of the reconstructed network using the ALoss differentiated loss function is higher than the MSE loss function. From Table IV, it can be seen that the SSIM value of the reconstructed network using the ALoss differentiated loss function is higher than that of RA-ACS_net+MSE. It can be seen that the use of the ALoss function proposed in this study can greatly improve the performance of the reconstructed network RA-ACS_net_V2. The ALoss loss function improves the PSNR, as well as significantly improves the SSIM, which indicates that the loss function not only effectively improves the visual quality of the image but also better maintains the structural features of the image during the reconstruction process. MSE, although it also performs well, has limited performance enhancement when the measurement ratio is high, and especially performs significantly worse than ALoss in PSNR.

At low measurement ratios (MR=0.04), the performances of the two loss functions are relatively close, but the advantage of ALoss is gradually revealed when the measurement ratio increases, especially at high measurement ratios (MR=0.5), its image recovery is significantly better than that of MSE. If the application scenarios require a better reconstruction effect at high measurement ratios, the ALoss loss function is recommended. MSE can also be an option if the basic performance at low measurement ratios is pursued, but it is not as good as ALoss in terms of comprehensive performance.

The ALoss loss function outperforms MSE at all measurement ratios, especially at high measurement ratios, where it significantly improves the quality of the image reconstruction and exhibits greater applicability and effectiveness. Future studies can further explore the improvement of this loss function to enhance its performance under low measurement ratio conditions.

TABLE III PSNR (DB) RESULTS FOR DIFFERENT LOSS FUNCTION DESIGN METHODS

Aarithmetic	MR=0.04	MR=0.1	MR=0.25	MR=0.5
RA-ACS_net+MSE	22.788	25.050	28.230	30.617
RA-ACS_net+ALoss	22.986	25.467	28.632	31.950

TABLE IV SSIM RESULTS FOR DIFFERENT LOSS FUNCTION DESIGN METHODS

Arithmetic	*MR=0.04*	*MR=0.1*	*MR=0.25*	*MR=0.5*
RA-ACS_net+MSE	0.713	0.814	0.896	0.927
RA-ACS_net+ALoss	0.713	0.833	0.914	0.950

C. Quantum Optical Bio-imaging System Experiments

In order to verify the validity of the RA-ACS_net application, the reconstruction performance of RA-ACS_net in quantum optical imaging is analysed in this section, and the specific results are shown in Fig. 17.

Fig. 17 gives the results of the comparison of the PSNR metrics of the reconstructed images of RA-ACS_net and the traditional iterative algorithm TAVL3 at different measurement rates (MR=0.04, 0.1, 0.25, 0.5). From Fig. 17, it can be seen that the reconstruction performance of both RA-ACS_net and TAVL3 is improved as the measurement ratio increases; the PSNR metrics of RA-ACS_net are improved by 3.265 dB~6.71 dB compared with TAVL3. It can be seen that RA-ACS_net is well-suited to be applied for reconstructing images in a photo-quantum optics bio-imaging system. The TVAL3 algorithm gradually improves the PSNR when the measurement ratio increases, but the overall performance is relatively flat, and finally, the PSNR is about 23 dB at a measurement ratio of 0.5. The Proposed algorithm improves the PSNR significantly faster, outperforms TVAL3 in all measurement ratio conditions, and the PSNR is close to 30 dB at a measurement ratio of 0.5.

At a lower measurement ratio (0.1), the PSNR of TVAL3 is about 20 dB, while that of the Proposed algorithm is about 24 dB, showing a significant improvement in the reconstruction quality of the newly Proposed algorithm at low measurement ratios. This difference suggests that the Proposed algorithm is more effective than TVAL3 in dealing with sparse data, and may be structurally designed to be more adept at extracting information from limited data. As the measurement ratio reaches 0.5, the PSNR of the Proposed algorithm reaches 30 dB, showing a strong reconstruction capability, while the PSNR of the TVAL3 algorithm is only 23 dB, which is still significantly lower than that of the Proposed algorithm. This indicates that the Proposed algorithm is able to make better use of the information and improve the accuracy of image reconstruction when more data are acquired.

The PSNR of the TVAL3 algorithm grows at a slower rate, especially in the range of measurement ratios from 0.2 to 0.5, and the PSNR of the Proposed algorithm is more linear with measurement ratio and grows at a higher rate, especially in the range of measurement ratios from 0.1 to 0.5.

The Proposed algorithm shows better image reconstruction capability than TVAL3 both at low and high measurement ratios, and is suitable for scenes that require high quality reconstruction. The performance of TVAL3, on the other hand,

improves more slowly and has greater limitations especially at low measurement ratios. In future research, we can continue to explore how to further optimize the Proposed algorithm to further improve the performance at higher measurement ratios.

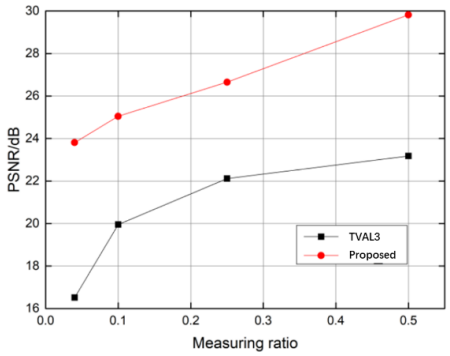


Fig. 17. Peak signal-to-noise ratio of reconstructed experimental maps with different algorithms.

VI. Conclusion and Outlook

Aiming at the problems such as the poor effect of image reconstruction methods in quantum optics bio-imaging systems, this study proposes a fusion quantum optics ultra-high resolution bio-imaging reconstruction method based on RA-ACS_net by combining the ripple algorithm and attention mechanism network. The method analyses the application of quantum optics technology in ultra-high resolution bio-imaging system and designs a quantum optics bio-imaging system, and at the same time explores and analyses the image reconstruction problem of quantum optics bio-imaging system, combines the attention mechanism network and the ripple algorithm, and puts forward the optimization of the ACS_net network model parameter method of the ripple algorithm, and constructs a bio-imaging reconstruction method based on the RA-ACS_net model. A bio-imaging reconstruction method based on the RA-ACS_net model is constructed. Comparative analysis of the proposed method using quantum optics-based bio-imaging image data shows that the RA-ACS_net method proposed in this study has a high peak signal-to-noise ratio for image reconstruction, and the image reconstruction effect is better. The subsequent work can consider denoising the image with noise reduction methods to improve the signal-to-noise ratio of the reconstructed image.

In the course of the study, we found that there are still several problems that have not been effectively addressed:

- Although the network proposed in this study has improved in the reconstruction effect, in practical applications, there is a certain noise interference in the quantum optical imaging process, especially in the low photon counting conditions, the noise may affect the imaging quality. How to further reduce the noise effect in the reconstruction process is still an urgent problem.
- The RA-ACS_net network increases the computational complexity while the accuracy is improved, especially when processing ultra-high resolution images, the reconstruction time is relatively long. Therefore, how to optimize the network structure, reduce the computational cost and improve the reconstruction speed is a direction to focus on in the future.
- The experiments in this study are mainly based on specific bioimaging datasets, which may behave differently on other types of bioimaging data, despite validating the effectiveness of the method. Future research needs to explore more diverse bioimaging scenarios and extend the generality of the network.

Based on the present and looking into the future, the following entry points can be studied: firstly, integrating noise reduction techniques, which can be combined with advanced noise reduction algorithms in the future, especially adaptive noise reduction methods based on deep learning, in order to further improve the signal-to-noise ratio of the reconstructed image, and to enhance the robustness of the system under low-signal conditions. The second is designing lightweight networks; future research can be devoted to designing more lightweight network models to reduce the consumption of computational resources and enhance the reconstruction speed. For example, methods such as pruning techniques or knowledge distillation can be introduced to optimise existing reconstruction networks. The third is the fusion of multimodal imaging. Considering the complexity of bio-imaging technology, the fusion of quantum optics with other imaging technologies (magnetic resonance, X-ray imaging) can be explored in the future to build a multimodal imaging system, which can provide richer biological information and achieve more accurate diagnosis and research. Finally, in order to further promote the practical application of the method, future research should validate the performance of the system in clinical or practical biological studies and assess its applicability and effectiveness in different scenarios.

References

[1] Kurosu J , Kanai K , Tsutsumi J .Label-free visualization of nano-thick biomolecular binding by electric-double-layer modulation[J].Sensors and Actuators B: Chemical, 2023, 382:133548.

[2] Bonifazi G , Barontini P , Gasbarrone D S S .Development of a Powder Analysis Procedure Based on Imaging Techniques for Examining Aggregation and Segregation Phenomena[J].journal of imaging, 2024, 10(3).

[3] Ahmed G , Hashim A .Synthesis and Tailoring Morphological and Optical Characteristics of PMMA/PEG/Si3N4 Hybrid Nanomaterials for Optics and Quantum Nanoelectronics Applications[J].Silicon, 2023.

[4] Yuan J , Zhang L , Kim C S .Multimodal Interaction of MU Plant Landscape Design in Marine Urban Based on Computer Vision Technology[J].Plants, 2023, 12(7).

[5] Yi W , Khalid A , Arshad N ,Asghar M S, Irshad M S, Wang X. Recent Progress and Perspective of an Evolving Carbon Family From 0D to 3D: Synthesis, Biomedical Applications, and Potential Challenges[J].ACS Applied Bio Materials, 2023, 6(6):2043-2088.

[6] Tripathi G , Guha L , Kumar H .Seeing the unseen: the role of bioimaging techniques for the diagnostic interventions in intervertebral disc degeneration[J].Bone Reports, 2024, 22.

[7] Parashar A K , Verma K K , Arora K V .A Concise Review of Carbon Dots and their Pharmaceutical and Biomedi-cal Applications[J].Recent Advances in Drug Delivery and Equationtion. 2023, 17(3):183-192.

[8] Chong K C , Pramanik M .Physics-guided neural network for tissue optical properties estimation.[J].Biomedical optics express, 2023, 146:2576- 2590.

[9] Wanas W , El-Kaream S A A , Ebrahim S , Soliman M, Karim M. Cancer bioimaging using dual mode luminescence of graphene/FA-ZnO nanocomposite based on novel green technique[J].Scientific reports, 2023, 13(1):27.

[10] Yao H , Dai F , Zhang S , Zhang Y, Tian Q, Xu C. DR2-Net: Deep Residual Reconstruction Network for Image Compressive Sensing[J].Neurocomputing, 2019, 359(sep.24):483-493.

[11] Sun Y, Chen J, Liu Q, et al. Dual-path attention network for compressed sensing image reconstruction[J]. IEEE Transactions on Image Processing, 2020, 29: 9482-9495.

[12] Meesala K , Rasala S .Potential of System of Rice Intensification (SRI) to Contribute to the Policy Objectives: Paradigm of Three-Tier Approach in Southern Telangana-A Case Study of Narayanpet[J].Agricultural Sciences, 2022, 13(4):542-554.

[13] Tinghui Wang, Zhenghua Gu, Tong Zhou, Yong Ma, Jiayi Zhang. Ripple algorithm and its application to the optimisation of Muskingum model parameters[J]. Journal of Water Resources and Water Transportation Engineering,2024,(02):81-90.

[14] Kowalska, A., & Michalska, M. Privacy-Preserving Federated Learning in Healthcare AI: A Comprehensive Review[J]. UKTIA Transactions on Emerging Technologies, 2023, 1–13.

[15] Li Z , Pan H , Shen G , Zhai D, Zhang W, Yang L. Single-photon Lidar for canopy detection with a multi-channel Si SPAD at 1064 nm[J].Optics & Laser Technology, 2023:157.

[16] Zhu Minhao,Ma Lei. Design of electronic payment protocol based on quantum entanglement measurement theory[J]. Journal of East China Normal University(Natural Science Edition),2024,(03):136-146.

[17] Lapointe J ,A. Grégoire,J. Bérubé, Vallée R. Enhancing Evanescent Wave Coupling of Near-Surface Waveguides with Plasmonic Nanoparticles[J]. Sensors (Basel, Switzerland), 2023, 23.

[18] FANG Lin, CAI Jian, YE Jialiang, ZHANG Qi. Diffraction modelling of digital micromirror devices and its application to laser direct writing machine[J]. Advances in Laser and Optoelectronics,2024,61(15):318-324.

[19] Ramanathan S , Ramasundaram M .Vector Quantized Convolutional Autoencoder for Low-Dose CT Image Reconstruction with Perceptual and Bias Reducing Loss[J].Journal of Uncertain Systems, 2023, 16(04).

[20] Fan H , Dong Q , Guo N .Surface defect classification of hot-rolled steel strip based on mixed attention mechanism[J].Robotic Intelligence and Automation, 2023(4):43.

[21] Fu Mengxi, Zhu Xiaoyu, Zhang Liang, Xu Chuanlong. Velocimetry particle field reconstruction method for optical field particle images based on deep residual neural network[J]. Journal of Optics,2024,44(16):152-162.

[22] Rashid A , Masood A , Khan A U R .ACS-IoT: Smart Contract and Blockchain Assisted Framework for Access Control Systems in IoT Enterprise Environment[J].Wireless Personal Communications, 2024, 136(3):1331-1352.

[23] LI Huiyuan, LEI Chunyang, HUANG Yan, NIE Zhou. Structural modification of fluorescent proteins and their biosensing applications[J]. Journal of Higher Education Chemistry,2020,41(11):2324-2334.

Privacy-Preserving Content-Based Medical Image Retrieval Using Integrated CNN Fusion and Quantization Optimization

Mohamed Jafar sadik[1], Muhammed E Abd Alkhalec Tharwat[2]*,
Dr. Noor Azah Samsudin[3], Dr. Ezak Fadzrin Bin Ahmad[4]
Faculty of Computer Science and Information Technology (FSKTM),
Universiti Tun Hussein Onn Malaysia (UTHM), Batu Pahat, Johor 86400, Malaysia[1, 3, 4]
Technical Engineering College, Al-Bayan University, Baghdad, Iraq[2]

***Abstract*—Content-Based Image Retrieval (CBIR) systems have become increasingly crucial in healthcare as the volume of medical imaging data continues to grow exponentially. However, existing systems struggle to balance privacy preservation, computational efficiency and retrieval accuracy, particularly in resource-constrained healthcare environments. This research proposes a novel multi-level privacy-preserving CBIR architecture that integrates multiple convolutional neural network (CNN) architectures with fusion strategies and quantization optimization specifically designed for encrypted medical images. The proposed framework addresses three key challenges: privacy preservation through advanced encryption techniques, feature extraction using optimized CNN fusion strategies and computational efficiency through model quantization. By implementing multiple pre-trained CNN models—including VGG-16, ResNet50, DenseNet121 and EfficientNet-B0—along with various fusion strategies, the system achieves improved feature extraction from encrypted medical images. The framework incorporates quantization techniques to optimize computational efficiency without compromising retrieval accuracy. Experimental results across multiple medical imaging modalities, including X-ray, magnetic resonance imaging (MRI) and computed tomography (CT) scans, demonstrate the effectiveness of the proposed approach in terms of retrieval accuracy, computational efficiency and security robustness. This research contributes to advancing privacy-preserving medical image analysis by providing a comprehensive solution that effectively balances security requirements with practical implementation constraints in healthcare settings.**

Keywords*—Content-Based Image Retrieval (CBIR); medical image analysis; privacy preservation; deep learning; convolutional neural networks (CNNs); feature fusion; model quantization; healthcare security; encrypted image processing; resource-constrained computing; computed tomography (CT); magnetic resonance imaging (MRI)

I. INTRODUCTION

In recent years, the healthcare industry has experienced an unprecedented surge in medical imaging data generated through various modalities including MRI, CT, and X-ray [1]. This exponential growth has created an urgent need for efficient Content-Based Image Retrieval (CBIR) systems that can accurately retrieve relevant medical images while preserving patient privacy. Current CBIR systems have demonstrated significant potential in healthcare applications, particularly with the integration of deep learning techniques [2, 3]. However, these systems often struggle to balance privacy preservation with retrieval efficiency, especially one in resource-constrained healthcare environments [4, 5].

The advent of deep learning, particularly Convolutional Neural Networks (CNNs), has revolutionized medical image analysis by enabling automatic extraction of complex hierarchical features [6]. Pre-trained models like VGG-16 have shown promise in medical applications, though they require significant adaptation when working with encrypted medical images [3, 4]. While existing research has explored various aspects of privacy-preserving CBIR systems, there is a notable absence of comprehensive solutions that integrate multiple CNN architectures with fusion strategies specifically designed for encrypted medical imaging domains [7, 8, 9].

Current literature reveals significant limitations in optimization approaches for computational efficiency in privacy-preserving medical CBIR systems. While some studies have investigated quantization and hashing techniques to enhance retrieval efficiency, there is insufficient research examining how these optimization methods affect the delicate balance between security, computational resources, and retrieval accuracy in encrypted medical image databases [10, 11]. This gap becomes particularly critical in resource-limited healthcare settings, where the computational burden of processing encrypted medical datasets can significantly impact system performance [4, 12].

Furthermore, existing encryption methods often fail to address the unique challenges posed by medical imaging, where preserving diagnostic information while ensuring patient privacy is crucial [13, 14, 15]. The integration of advanced encryption techniques with optimized deep learning models remains largely unexplored, particularly in developing systems that can maintain both security and accuracy while operating within the computational constraints of typical healthcare environments [5, 16, 17]. This gap in research highlights the need for a comprehensive framework that effectively combines privacy preservation mechanisms, efficient deep learning techniques, and optimization methods suitable for real-world healthcare applications.

*Corresponding Author.

The lack of integrated approaches combining multiple CNN architectures, fusion strategies, and quantization techniques specifically designed for encrypted medical image retrieval represents a significant research opportunity. While individual components have been studied separately, there is a clear need for a unified framework that can effectively balance privacy preservation, computational efficiency, and retrieval accuracy [10, 11].

Dataset considerations: Most prior studies evaluate privacy-preserving CBIR systems on a single imaging modality or on datasets of limited size, which makes it difficult to assess generalization across clinical scenarios. To address this limitation, our work utilizes a balanced multi-modal dataset comprising bone X-rays, chest radiographs and brain MRI scans drawn from widely used collections such as MURA, Chest X-ray and BTTypes. By explicitly indicating the choice of datasets and splitting them into training, validation and test sets, we provide a transparent experimental foundation for evaluating the proposed framework.

The remainder of this study is structured as follows: The primary contributions of the work are consolidated in Section II, where we highlight the novel aspects and implications of our study. Section III presents the related work and literature survey. In Section IV, we describe the proposed methodology in detail, including the foundational design choices and theoretical formulation. Sections V and VI presents and discuss the experimental setup, dataset descriptions, performance metrics, and key results obtained from our approach. Finally, Section VII offers concluding remarks, summarizing the main findings and outlining prospective directions for future research.

II. Contributions

This study presents several contributions aimed at advancing privacy-preserving medical image retrieval systems:

1) A novel multi-level privacy-preserving CBIR architecture that integrates block-wise encryption with secure feature extraction, providing superior security and computational efficiency compared to traditional single-level approaches in medical image retrieval systems.

2) A comprehensive framework for implementing multiple pre-trained CNN architectures with fusion strategies, specifically optimized for encrypted medical images, which significantly improves feature extraction accuracy compared to single-model approaches while maintaining privacy requirements. Advanced Approach to Quantization.

3) An innovative model quantization framework that substantially reduces computational overhead while maintaining high retrieval accuracy in resource-constrained healthcare environments, enabling practical deployment of privacy-preserving medical image retrieval systems.

III. Related Work and Literature Survey

The field of privacy-preserving medical image retrieval has progressed rapidly over the past two decades. Early approaches focused on hand-crafted features extracted from encrypted images and relied on distance-preserving transformations or homomorphic encryption to enable similarity search without exposing raw data [18, 19, 20]. Subsequent works introduced secure index structures and hashing schemes such as secure kNN, deep hashing and bag–of–encrypted words to improve retrieval speed and accuracy [21, 22]. More recently, deep learning–based solutions have emerged that employ convolutional neural networks (CNNs) trained on encrypted feature representations, often combined with federated learning or homomorphic encryption to protect patient privacy [23, 24, 25]. These works demonstrate the potential of deep models but typically utilise a single architecture and do not address the trade–off between computational efficiency and retrieval accuracy when working with encrypted medical images. Recent literature also highlights the importance of secure cloud-based infrastructures. Yadav and Chokkalingam [26] proposed a two–step cloud-based CBIR system that combines encryption with watermarking and a principal component analysis (PCA) based feature extraction pipeline. Their framework first encrypts images and embeds watermark bits to trace unauthorised duplication; only authenticated users can decrypt and extract the watermark, enabling traceability of data misuse. Feature vectors are derived using a dominant local pattern and PCA, and retrieval is performed using these compact representations. The authors demonstrate that the two-step approach improves mean average precision and recall while maintaining robustness against unauthorised access. While this method enhances security in a cloud environment, it still relies on hand-crafted features and does not explore model fusion or quantization for computational efficiency.

A. Comparative Analysis of Datasets

Table I summarises several widely used open medical imaging datasets alongside the custom dataset employed in this study. The table emphasises key characteristics such as the imaging modality, number of images or studies, number of patients, available labels, and resolution.

As shown in Table I, public datasets such as MURA and ChestX-ray14 provide large numbers of radiographic images and detailed annotations, but focus on single modalities. The BTTypes dataset offers balanced benign and malignant brain MRI scans but is limited to a binary classification task. In contrast, our study employs a multi–modal dataset comprising X-ray, computed tomography (CT) and magnetic resonance imaging (MRI) images, each category containing 1,200 images. By combining bone, chest and brain imaging modalities, the experiments assess the generality of the proposed retrieval framework across diverse anatomical regions while retaining a manageable dataset size for in-depth analysis.

TABLE I. COMPARISON OF WIDELY USED MEDICAL IMAGING DATASETS WITH THE DATASET UTILIZED IN THIS WORK

Dataset	Modality	Images / Studies	Patients	Labels / Classes	Notes
MURA [27]	X-ray (upper extremity)	40,561 images from 14,863 studies	12,173	Normal/abnormal	Seven body parts (elbow, finger, forearm, hand, humerus, shoulder, wrist); images collected between 2001–2012.
ChestX-ray14 [28]	Chest X-ray	112,120 frontal images	32,717	Up to 14 thoracic pathologies	Resolution 1024×1024 px; patient-wise split of 86,524 training and 25,596 test images.
BTTypes (Brain tumor) [29]	Brain MRI	2,400 images	—	Benign / malignant	Two collections containing 1,200 benign and 1,200 malignant images were used for experiments.
This work	X-ray, CT and MRI	3,600 images	—	Bone, chest, brain	1,200 images per category; images encrypted using multi-tiered texture and color encryption prior to training and evaluation.

B. Key Takeaways

The existing literature underscores several research gaps that motivate our contributions. First, many privacy-preserving CBIR systems rely on a single CNN architecture or hand-crafted features and do not leverage the complementary strengths of multiple models. Second, computational efficiency and model compression are often overlooked, limiting the practical deployment of secure retrieval systems in resource-constrained environments. Third, comparative analyses of datasets are rarely provided, leaving unclear how performance generalizes across different modalities and imaging conditions. The proposed framework addresses these limitations by integrating multi-level encryption, multi-model CNN fusion, and quantization within a unified architecture while evaluating the system on a balanced multi-modal dataset.

IV. METHODOLOGY

Our methodology establishes a comprehensive framework for privacy-preserving content-based image retrieval (CBIR) in medical applications, addressing the critical challenges of security, accuracy, and computational efficiency. This multi-faceted approach integrates three key components: 1) a novel multi-level encryption technique designed specifically for medical images, 2) a robust feature extraction process utilizing multiple complementary CNN architectures, and 3) an optimized model quantization strategy to enhance computational efficiency while maintaining high retrieval accuracy. Each component is mathematically formalized and algorithmically implemented to ensure a systematic and reproducible approach. The following subsections detail the problem formulation, system architecture, and specific implementations of each component, demonstrating how they collectively create a balanced solution for secure and efficient medical image retrieval in resource-constrained healthcare environments.

A. Problem Formulation

Let I = {i1, i2, ..., in} be a set of medical images in the database. The multi-level encryption process can be defined as:

$$E = \{e_1, e_2, \ldots, e_n\} = \{\mathcal{E}(i_k, K_k)\}_{k=1}^{n} \quad (1)$$

where, E is the encryption function and K_k is the encryption key for image i_k. The block-wise encryption process divides each image into blocks B = {b1, b2, ..., bm} and applies local and global scrambling:

$$\mathcal{E}(i, K) = \mathcal{G}\left(\{\mathcal{L}(b_j, K)\}_{j=1}^{m}\right) \quad (2)$$

where, $\mathcal{L}$ represents local block encryption and $\mathcal{G}$ represents global scrambling.

Given the encrypted images E, the feature extraction process using multiple CNNs can be formulated as:

$$F = \{\Psi_1(e), \Psi_2(e), \ldots, \Psi_k(e)\}, e \in E \quad (3)$$

where, Ψi represents the i-th CNN model (VGG-16, ResNet50, etc.). The fusion strategy combines these features:

$$F_{\text{fused}} = \omega_1 F_1 \oplus \omega_2 F_2 \oplus \ldots \oplus \omega_k F_k \quad (4)$$

where, ω_i are fusion weights and $\oplus$ represent the fusion operation.

Let M be the original model with weights W. The quantization problem can be formulated as:

$$\min_{W_q} \|W - W_q\|_2 \text{ subject to } W_q \in \{-2^{b-1}, \ldots, 2^{b-1} - 1\} \quad (5)$$

where, W_q represents the quantized weights and b is the bit width. The overall optimization objective combines all three components:

$$\max_{\mathcal{E},\Psi,W_q} \left\{ \text{Accuracy}(R \mid Q) \text{ subject to: } \begin{cases} \text{Privacy}(E) \geq \tau_p \\ \text{Memory}(W_q) \leq \tau_m \\ \text{Compute}(\Psi) \leq \tau_c \end{cases} \right\}. \quad (6)$$

where, R is the retrieved result set, Q is the query image, τ_p is the privacy threshold, τ_m is the memory threshold, and τ_c is the computational threshold. This formulation encapsulates the key challenges of maintaining privacy through secure encryption, ensuring accurate feature extraction and retrieval, and optimizing computational efficiency through quantization, while balancing these competing objectives within practical constraints.

B. System *Architecture*

The system architecture encompasses a comprehensive privacy-preserving framework for medical image retrieval, structured around three main entities: Data Owner, Cloud Server and Query User. Each entity performs distinct roles within an integrated workflow that ensures both security and efficiency. The Data Owner is responsible for system initialization, including encryption key management, medical image preprocessing and secure feature extraction. This entity encrypts both the medical images and their corresponding feature vectors before uploading them to the Cloud Server. The Cloud Server functions as a secure storage and processing unit, maintaining

the encrypted database while performing similarity searches on encrypted feature vectors without accessing the raw data. A high-level overview of the sequence of operations between these entities is illustrated in Fig. 1.

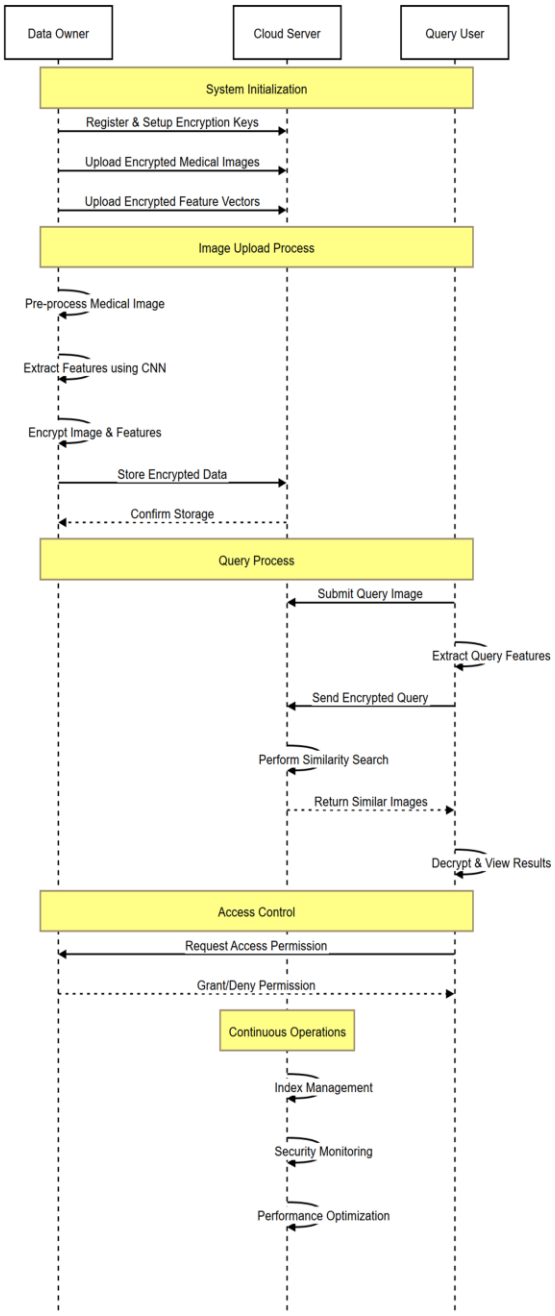


Fig. 1. System architecture sequence diagram.

The operational workflow begins with the Data Owner who manages the original dataset and encryption keys, while also providing training data from the same distribution for model training. During the initialization phase, the Data Owner implements a multi-tiered texture and color encryption (MTTCE) algorithm that preserves crucial diagnostic information while securing patient privacy. This process combines local and global image scrambling techniques to ensure that encrypted images maintain their utility for retrieval purposes. Following encryption, the Data Owner implements the feature extraction process using multiple pre-trained CNN architectures—VGG-16, ResNet50, DenseNet121, and EfficientNet-B0—each specifically modified and fine-tuned for encrypted medical image analysis.

The Cloud Server component maintains both the encrypted image database and the trained feature extraction models. Upon receiving an encrypted query image, the server extracts features using the optimized models and computes similarities with database features using Euclidean distance metrics. The server then returns a set of most similar encrypted images based on feature matching. Throughout this process, the Cloud Server operates exclusively within the encrypted domain, ensuring that sensitive medical information remains protected during all computational operations.

The Query User interacts with the system by submitting query images, which undergo similar preprocessing and encryption before a similarity search is performed. After receiving encrypted result sets, the Query User must request corresponding decryption keys from the Data Owner to access the original medical images. This additional layer of security and access control ensures that only authorized users can view decrypted content. The entire architecture employs robust access control mechanisms to guarantee that only authenticated users can interact with the system, with multi-factor authentication and session management protocols enhancing security throughout the retrieval process.

This architecture is designed with particular attention to efficiency in resource-constrained healthcare environments. Through this comprehensive design, the system architecture successfully balances the competing requirements of privacy preservation, computational efficiency, and retrieval accuracy in medical image analysis. Fig. 2 illustrates the complete framework architecture of our proposed system.

C. Multi-Level Encryption Approach

The encryption process in our system employs a multistage homomorphic encryption method specifically designed to securely store medical images while maintaining their utility for retrieval. This approach effectively preserves both local and global image information by encoding them into a binary string, ensuring that query-relevant features remain accessible while protecting against unauthorized access and information leakage.

The encryption process consists of three main phases. First, the Local Texture Protection phase segments the original image into non-overlapping subblocks and scrambles the RGB channel values within each subblock, effectively obscuring local texture information while preserving features necessary for retrieval. Second, the Global Texture Preservation phase randomly swaps positions between subblocks, disrupting the global image structure while maintaining essential texture information for retrieval purposes. Third, the Color Information Security phase involves channel swapping and RGB value substitution for each scrambled subblock, creating a non-linear encryption model by binding substitution values to the positions of encrypted sub-blocks.

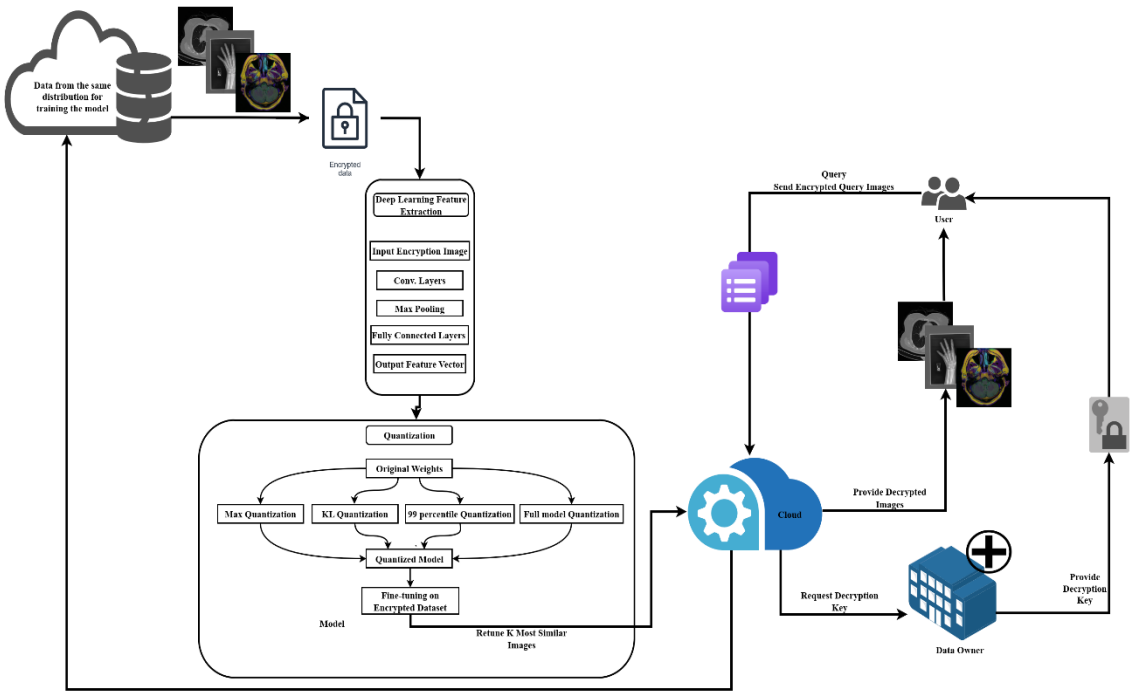


Fig. 2. Proposed framework architecture.

The encryption key is generated using a built-in random function and is essential for authorized decryption. The final encrypted image is composed by combining the processed subblocks, resulting in a secure representation that maintains no visible relationship to the original content while preserving the features necessary for effective retrieval. This process is formalized in Algorithm 1, which outlines the complete multi-stage encryption approach.

D. Deep Feature Extraction with Multiple CNN Models

Our system leverages multiple pre-trained CNN architectures for robust feature extraction from encrypted medical images. We selected VGG16, ResNet50, DenseNet121, and EfficientNetB0 for their proven effectiveness in image feature extraction tasks [30, 31, 32]. Each architecture was strategically adapted for our privacy-preserving CBIR system through a systematic fine-tuning process.

Each CNN architecture underwent several modifications to optimize its performance with encrypted medical images. Initial weights were obtained from pre-training on the ImageNet dataset, providing a strong foundation for general feature extraction capabilities. The models were then fine-tuned using our encrypted medical image dataset to adapt to the specific characteristics of encrypted data. To optimize computational efficiency while maintaining feature quality, we implemented selective layer freezing, where all layers except the final 10 were frozen during training. The architecture was enhanced by incorporating a Global Average Pooling (GAP) layer, which reduces spatial dimensions while preserving channel information followed by a task-specific classification head.

The feature extraction process for an input image can be mathematically formalized as:

$$F(I) = h\big(g(f_\theta(I))\big) \tag{7}$$

where, $f_\theta(I)$ represents the frozen pre-trained layers with parameters θ, $g()$ denotes the Global Average Pooling operation, and $h()$ represents the task-specific classification head. This architecture ensures effective feature extraction while maintaining compatibility with our privacy-preserving framework.

E. Fusion Strategies for Enhanced Performance

To leverage the complementary strengths of different CNN architectures, we implemented three fusion strategies: attention-based fusion, weighted average ensemble, and majority voting. These fusion methods combine the feature representations or predictions from multiple models to achieve more robust and accurate results than any single model could provide. We selected these specific fusion strategies based on their theoretical foundations and empirical performance in medical image analysis tasks. Attention-based fusion was chosen for its ability to adaptively weight features according to their relevance for each specific image, addressing the high variability in medical image characteristics. The weighted average ensemble was included for its computational efficiency and proven effectiveness in scenarios where model strengths remain relatively consistent across a dataset. Majority voting was selected as a robust decision-level fusion technique that can effectively mitigate the impact of individual model failures or inconsistencies [33, 34, 35].

The attention-based fusion approach applies an attention mechanism to dynamically weight the contributions of different models based on the input image characteristics. This method learns to focus on the most relevant features from each model, enhancing the overall representation quality. The fusion process can be expressed as:

$$F_{\text{fused}} = \sum_{i=1}^{k} \alpha_i(I) \cdot F_i(I) \tag{8}$$

where, $\alpha_i(I)$ represents the attention weight for model i given input image I, and $F_i(I)$ is the feature representation from model i.

The weighted average ensemble assigns fixed weights to each model based on their validation performance, combining their outputs in a predetermined ratio. This approach provides a simple yet effective method for model fusion when the relative strengths of different models are known in advance.

The majority voting strategy operates at the decision level, combining the class predictions from multiple models to determine the final classification. This approach is particularly effective for reducing the impact of outlier predictions from individual models.

These fusion strategies enhance the robustness and accuracy of our system, particularly when dealing with the challenging task of feature extraction from encrypted medical images. The complementary nature of different CNN architectures allows our system to capture a wider range of relevant features, improving retrieval performance across diverse medical image types.

F. Model Quantization for Computational Efficiency

Our system implements four distinct quantization approaches to balance model efficiency with accuracy: max quantization, KL divergence–based quantization, 99th percentile quantization and full model quantization. Each method offers different trade-offs between compression ratio and model performance. The relationship between these quantization modes and the broader workflow of our privacy-preserving CBIR system is visualized in Fig. 3, which depicts the comprehensive processing pipeline from encryption through feature extraction and quantization to retrieval.

Max quantization scales the weights based on the maximum absolute value in each layer:

$$W_{\text{quantized}} = \frac{W}{max(|W|)} \times (2^b - 1) \quad (9)$$

where, W represents the original weights, b is the target bit-width (typically 8 bits), and $max(|W|)$ is the maximum absolute value in the weight tensor. KL divergence-based quantization optimizes the quantization thresholds by minimizing the KullbackLeibler divergence between the original and quantized weight distributions:

$$KL(P\|Q) = \sum_i P(w_i)\log\frac{P(w_i)}{Q(w_i)} \quad (10)$$

where, P represents the distribution of original weights and Q represents the distribution of quantized weights.

99th percentile quantization uses the 99th percentile of absolute weight values as the quantization threshold instead of the maximum value:

$$W_{\text{quantized}} = \frac{W}{\text{percentile}_{99}(|W|)} \times (2^b - 1) \quad (11)$$

Full model quantization applies integer quantization to all layers of the model, including weights, activations, and biases, following the equations:

$$W_{\text{int }8} = \text{round}\left(\frac{W}{S_w}\right), A_{\text{int8}} = \text{round}\left(\frac{A}{S_a}\right),$$
$$B_{\text{int }32} = \text{round}\left(\frac{B}{S_w \times S_a}\right) \quad (12)$$

where, S_w and S_a are scaling factors for weights and activations. The implementation of these quantization methods is formalized in Algorithm 3, which outlines the complete quantization process for different approaches.

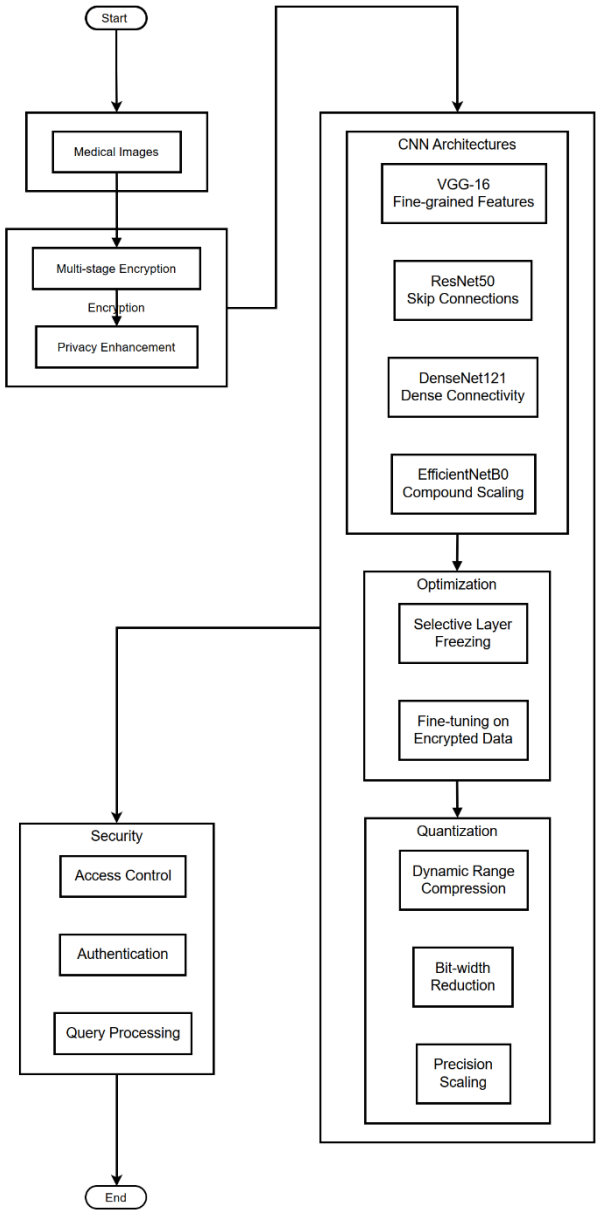


Fig. 3. Comprehensive workflow of the proposed privacy-preserving medical image retrieval system.

G. Algorithms

The algorithmic foundation of the privacy-preserving CBIR system encompasses three primary components: multi-level encryption, feature extraction using multiple CNN architectures, and model optimization through quantization. Each component incorporates specialized algorithms designed to enhance

security, accuracy, and efficiency within the medical image retrieval context. The encryption process employs a multi-stage homomorphic encryption method formalized in Algorithm 1, which effectively preserves both local and global image information while protecting against unauthorized access.

Algorithm 1: Multi-Stage Image Encryption for CBIR Systems

Require: Original Image I

Ensure: Encrypted Image I_{enc}, Encryption Key K

1. Divide the original image I into non-overlapping subblocks B_i
2. **for** each subblock B_i **do**
3. Scramble the positions of the RGB channel values within B_i to hide local texture information
4. **end for**
5. Randomly scramble the positions between subblocks B_i to obscure global texture information
6. **for** each scrambled subblock B_i **do**
7. Substitute the RGB channel values and swap the channels to secure global color information
8. **end for**
9. Generate the encryption key K using a built-in random function
10. Combine the processed subblocks to form the encrypted image I_{enc}

The feature extraction process utilizes multiple pretrained CNN architectures that have been specifically adapted for working with encrypted medical images. Algorithm 2 formalizes the feature extraction process using these adapted models:

Algorithm 2: Multi-Model Feature for Encrypted Medical Images

Require: Encrypted Image I_{enc}, Set of CNN Models M=$\{m_1, m_2, \ldots, m_k\}$

Ensure: Feature Vector Set $F=\{f_1, f_2, \ldots, f_k\}$

1: **for** each model $m_i \in M$ **do**

2: Preprocess I_{enc} according to model requirements

3: Extract feature maps X_i from convolutional layers of m_i

4: Apply Global Average Pooling: $g_i = \frac{1}{H \times W} \sum_{h=1}^{w} \sum_{w=1}^{w} X_{i,h,w}$

5: Apply model-specific transformation: $f_i = \sigma(w_{igi} + b_i)$

6: **end for**

7: **return** Feature vector set F

The model optimization framework implements four distinct quantization approaches, formalized in Algorithm 3, which balance model efficiency with accuracy:

Algorithm 3: Model Quantization Framework

Require: Trained model M with weights W quantization mode *mode*, target bit-width b

Ensure: Quantized model M'

1. **If** *mode* == "max" **then**
2. $W_{max} \leftarrow \max(|W|)$
3. *Quantize* weights: $W_q = \frac{W}{W_{max}} \times (2^b - 1)$
4. **else if** *mode* == "KL" **then**
5. Calculate weight histogram $H(W)$
6. **for** each potential threshold t **do**
7. Quantize weights using threshold t
8. Calculate KL divergence $KL(P||Q_t)$
9. **end for**
10. Select optimal threshold $t_{opt} = \arg\min_t KL(P||Q_t)$
11. Quantize weights using t_{opt}
12. **else if** *mode* == "99%" **then**
13. Calculate 99th percentile p_{99} = precentile$_{99}$ (|W|)
14. Quantize weights: $W_q = \frac{W}{p_{99}} \times (2^b - 1)$
15. **else if** *mode* == "full" **then**
16. Calculate scaling factors $S_w = \frac{\max(|W|)}{127}$, $S_a = \frac{\max(|A|)}{127}$
17. Quantize weights: W_{int8} = round $(\frac{W}{S_w})$
18. Quantize activations: A_{int8} = round $(\frac{A}{S_a})$
19. Quantize biases: B_{int32} = round $(\frac{B}{S_w \times S_a})$
20. **end if**
21. Replace original weights with quantized weights
22. **Return** quantized model M'

The integration of these algorithms creates a complete workflow for privacy-preserving medical image retrieval, formalized in Algorithm 4:

Algorithm 4: Privacy-Preserving Medical Image Retrieval Workflow

Require: Medical image database I, query image Q, encryption key set K, quantization mode *mode*

Ensure: Retrieved similar images R

1. Encrypt all database images: $E = \{\mathcal{E}(i_k, K_k)\}_{k=1}^{n}$
2. Train and optimize CNN models $M = \{m_1, m_2, \ldots, m_k\}$ on encrypted images
3. **for** each model $m_i \in M$ ***do***
4. Apply quantization using selected *mode*
5. **end for**
6. Extract and store features from all encrypted database images
7. Encrypt query image: $EQ = \mathcal{E}(Q, k_q)$
8. Extract query features using quantized models
9. Compute similarities between query features and database features
10. Return top-k most similar encrypted images ER
11. Decrypt results: $R = \{D(er_j, k_j)\}_{j=1}^{k}$

These algorithms work in concert to create a system that effectively balances privacy preservation, computational efficiency, and retrieval accuracy. The multistage encryption described in Algorithm 1 ensures security while preserving features necessary for retrieval. The feature extraction process detailed in Algorithm 2 leverages multiple CNN architectures to provide robust feature extraction from encrypted data. The quantization framework formalized in Algorithm 3 optimizes models for deployment in resource-constrained environments without significantly compromising retrieval performance. The complete workflow presented in Algorithm 4 integrates all

components into a cohesive privacy-preserving medical image retrieval system.

H. Dataset

To evaluate the proposed framework across multiple imaging modalities, we curated a balanced dataset comprising three categories: bone X-rays, chest radiographs and brain MRI scans. Each category contained 1,200 images, yielding a total of 3,600 images. The images were acquired from publicly available collections and internal sources. As summarized in Table I, publicly available datasets such as MURA and ChestX-ray14 contain tens of thousands of images but focus on a single modality. The BTTypes dataset contains 2,400 brain tumor MRI images split evenly between benign and malignant cases. For our experiments we combined data from these sources and additional CT images to obtain a balanced set across modalities.

1) Dataset organization: Images were stratified by anatomical region to form three classes:

a) Bone X-ray: 1,200 multi-view musculoskeletal radiographs sampled from the MURA dataset [27]. Each study comprises projections of upper extremity bones labelled as normal or abnormal, and we ensured balanced representation across bone types.

b) Chest radiograph: 1,200 frontal chest X-ray images drawn from the ChestX-ray14 dataset [28]. Images cover up to 14 thoracic pathologies; we sampled a representative subset across different disease labels and included normal cases.

c) Brain MRI: 1,200 T1-weighted brain MRI scans from the BTTypes dataset [29] comprising equal numbers of benign and malignant tumor cases. Additional normal MRI scans were included to form a three-class problem (benign, malignant and healthy).

The dataset was divided into training, validation and testing subsets using a 1,000:100:100 split for each class. This partitioning ensures sufficient samples for model training while retaining disjoint validation and test sets for unbiased evaluation. Stratification by modality prevents data leakage across splits and allows separate assessment of each anatomical region.

2) Data processing pipeline: All images underwent the following processing steps to prepare them for encrypted retrieval:

a) Preprocessing: Images were resized to 224×224 pixels and normalized to match the input requirements of the pre-trained CNNs.

b) Encryption: Each image was encrypted using the multi-tiered texture and color encryption (MTTCE) scheme described in Section IV. Both plaintext and encrypted versions were retained for comparative experiments.

c) Feature extraction: Encrypted images were processed by the VGG-16, ResNet50, DenseNet121 and EfficientNet-B0 architectures to extract deep features. Features from the penultimate layer were used for classification and retrieval.

d) Quantization evaluation: Features were quantized using the four approaches outlined in Section IV (max, KL divergence, 99th percentile and full quantization) to assess the effect on accuracy and model size.

Maintaining a consistent pipeline for each modality allowed fair comparison across encryption and quantization settings. The balanced dataset and stratified splits ensure reproducibility and facilitate analysis of the proposed methods on diverse medical imaging data.

V. Experimental Results and Analysis

A. Experimental Design

The experimental evaluation was conducted with particular attention to reproducibility and practical applicability in medical imaging environments. The system was implemented using Python 3.8 with PyTorch 1.9 as the primary deep learning framework. All experiments were performed on a workstation equipped with an NVIDIA RTX 3080 GPU with 10GB VRAM, supported by 32GB system RAM and an Intel i7 processor. This hardware configuration was chosen to represent a realistic deployment environment while providing sufficient resources for effective model training and evaluation.

The implementation framework incorporated several key components, including PyTorch for model development, OpenCV and PIL for image preprocessing, TensorFlow's optimization toolkit for model quantization, and custom evaluation scripts developed using scikit-learn and NumPy for computing performance metrics. The system configuration parameters were carefully selected based on preliminary experiments and established best practices in medical image analysis. These parameters included image dimensions of 224×224 pixels, batch size of 32, and normalization parameters aligned with standard practices for pre-trained models.

The experimental evaluation utilized three distinct medical imaging datasets, each comprising 1,200 images systematically categorized based on anatomical regions: bone images, chest radiographs, and brain MRI scans. The datasets were strategically segregated into training, validation, and testing sets using a 1000:100:100 split ratio, ensuring robust model training while maintaining sufficient data for validation and testing. This data organization enabled direct comparative analysis between different encryption and quantization approaches while evaluating system performance.

Each image in the dataset underwent a systematic processing pipeline, including preservation of the initial plaintext format, application of Multi-Tiered Texture and Color Encryption (MTTCE), feature extraction using the various CNN architectures, and analysis under different quantization schemes. This standardized approach to data organization and processing ensured the reliability and reproducibility of the experimental results across all testing scenarios.

The experimental evaluation was structured to systematically address three main research objectives: evaluating the multi-level privacy-preserving CBIR architecture, assessing multiple CNN architectures and fusion strategies for feature extraction, and analyzing the model optimization framework based on quantization techniques. Each objective was evaluated using specific metrics tailored to measure the relevant aspects of system performance. Privacy

preservation was assessed using standard cryptographic metrics including entropy, Mean Squared Error (MSE), Structural Similarity Index (SSIM), and Peak Signal-to-Noise Ratio (PSNR). Model performance was evaluated using classification accuracy, precision, recall, and F1-score, while computational efficiency was measured through model size reduction and inference time metrics. This comprehensive experimental design enabled a systematic and thorough evaluation of all aspects of the proposed privacy-preserving CBIR system, ensuring that the results provided valuable insights into the practical implementation of secure and efficient medical image retrieval systems.

B. Result Analysis

The privacy preservation capabilities were evaluated through standard cryptographic metrics, comparing the proposed approach against the AES standard, as shown in Table II. The encryption algorithm achieved an entropy value of 7.392, approaching the theoretical optimal value of 8, indicating a high degree of randomness in the encrypted output. The Mean Squared Error (MSE) analysis revealed the algorithm achieved a higher distortion (12,766.02) compared to AES (2,625.48), indicating stronger alteration of the original image content. This higher MSE suggests enhanced security through greater deviation from the original image structure. The Structural Similarity Index Measure (SSIM) further validates the algorithm's effectiveness, with a value of 0.273204 compared to AES's 0.00283838. While AES shows a lower SSIM value, indicating nearly complete structural dissimilarity from the original image, our algorithm maintains a strategic balance between security and utility, preserving just enough structural information to enable effective feature extraction while still sufficiently obfuscating sensitive patient data. Additionally, the Peak Signal-to-Noise Ratio (PSNR) of 7.07 for our algorithm, lower than AES's 13.94, confirms the significant distortion introduced by our approach. A lower PSNR value is desirable for encryption algorithms as it indicates greater divergence from the original signal, making unauthorized reconstruction more difficult. This combination of metrics demonstrates that our algorithm achieves robust privacy protection while maintaining the utility necessary for content-based medical image retrieval tasks.

TABLE II. COMPARATIVE EVALUATION OF ENCRYPTION METRICS: OUR ALGORITHM VERSUS AES

Metric	Our Algorithm	AES
Entropy	**7.392129887**	1.4426951601859516e-10
MSE	**12766.02**	2625.48
SSIM	0.273204	**0.00283838**
PSNR	**7.070247655**	13.93870288

These metrics demonstrate the effectiveness of the encryption scheme in protecting sensitive medical image data while maintaining feature extractability.

The evaluation of CNN architectures for feature extraction revealed distinct patterns in their learning dynamics and classification performance. VGG-16 demonstrated exceptional performance with training accuracy reaching 98.7% and validation accuracy peaking at 98.8%, showing the most stable learning progression among all models. Its confusion matrix revealed strong classification capabilities with 1,468 correct bone classifications and 897 correct MRI classifications, achieving an impressive overall accuracy of 99%, as shown in Fig. 4 and Table III.

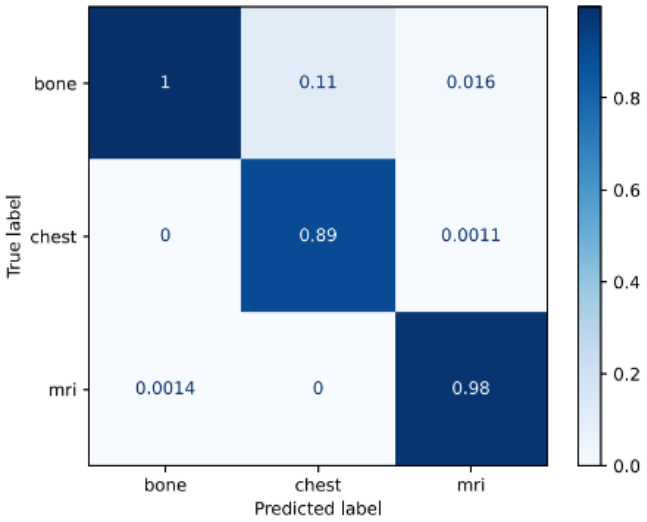


Fig. 4. Confusion matrix for the VGG-16 model showing classification performance across three medical image categories (bone, chest, MRI).

TABLE III. CLASSIFICATION PERFORMANCE METRICS FOR VGG-16 MODEL

Class	Precision	Recall	F1-score	Support
Bone	1.00	0.98	0.99	1500
Chest	0.89	0.99	0.94	97
MRI	0.98	1.00	0.99	900
Accuracy			0.99	2497
Macro avg.	0.96	0.99	0.97	2497
Weighted avg.	0.99	0.99	0.99	2497

ResNet50, while achieving comparable final performance with 1,465 correct bone classifications and 895 correct MRI predictions as illustrated in Fig. 6, exhibited more volatile learning behavior, particularly in early epochs where validation accuracy fluctuated between 98% and 92%. The learning curves for ResNet50 are shown in Fig. 5.

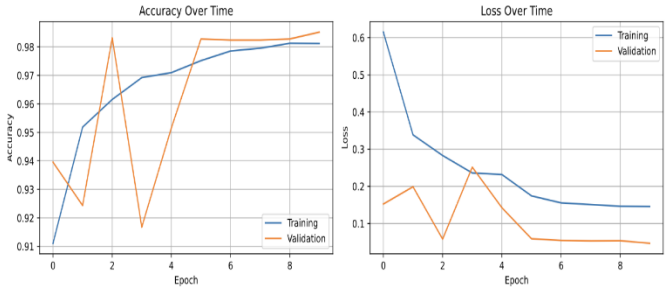


Fig. 5. Learning curves showing training and validation accuracy (left) and loss (right) over training epochs for the ResNet50 model.

DenseNet121, as given in Fig. 7, showed remarkable initial performance with validation accuracy starting at 97% and

maintaining consistent improvement to exceed 98.5%, demonstrating efficient feature extraction capabilities from the outset, as shown in Fig. 7 and Fig. 8.

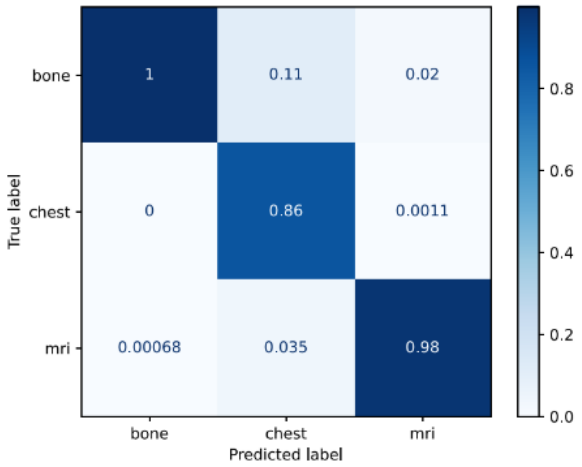


Fig. 6. Confusion matrix for ResNet50 model displaying classification results for bone, chest, and MRI images.

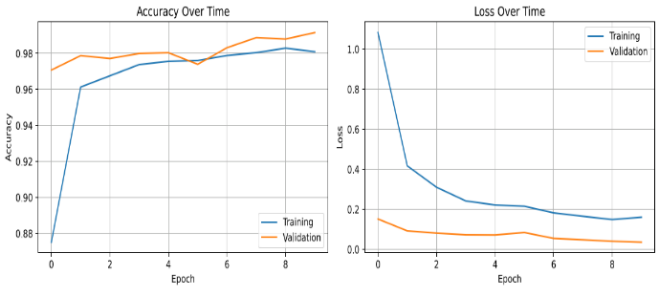


Fig. 7. Training and validation metrics for the DenseNet121 model across epochs.

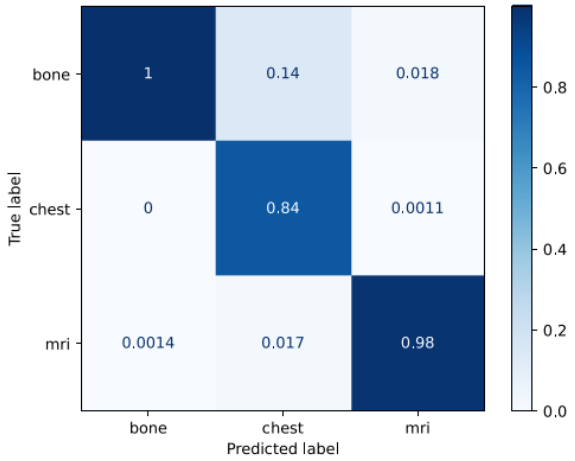


Fig. 8. Confusion matrix for DenseNet121 model showing the distribution of predictions across medical image types.

In contrast, EfficientNet-B0 as given in Fig. 9 showed more modest performance metrics, with training accuracy reaching 85% and experiencing early training instability, though it eventually stabilized at 83 to 84%, as illustrated in Fig. 9 and Fig. 10.

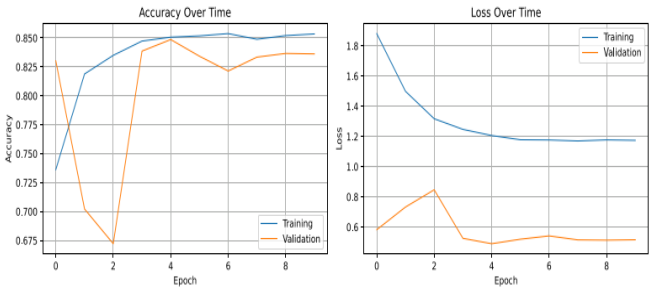


Fig. 9. Training and validation metrics for the EfficientnetB0 model across epochs.

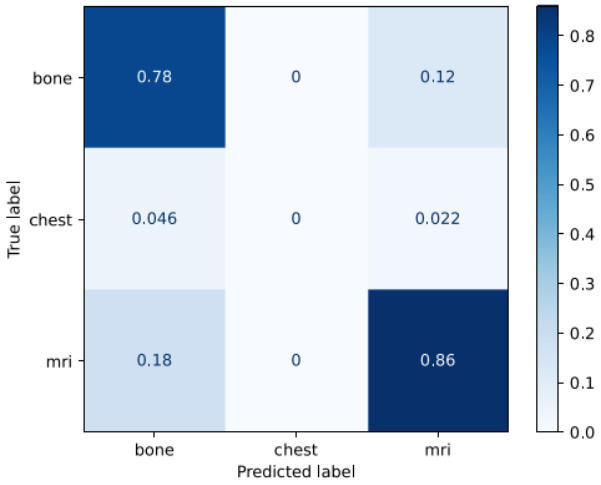


Fig. 10. Confusion matrix for the EfficientNetB0 model displaying classification results.

The fusion strategies, implemented to combine the strengths of individual CNN models, showed promising results. The attention-based fusion approach demonstrated superior performance compared to both individual models and other fusion strategies, achieving the highest overall accuracy at 99.52%, as shown in Fig. 11 and Fig. 12.

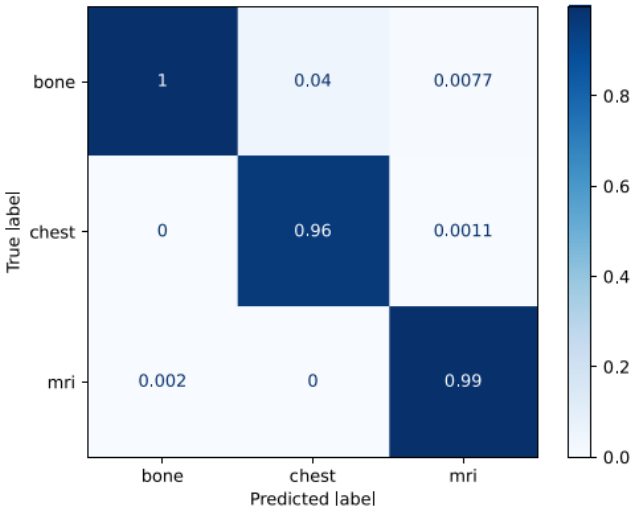


Fig. 11. Confusion matrix showing the classification performance of the attention-based fusion model across bone, chest, and MRI image categories.

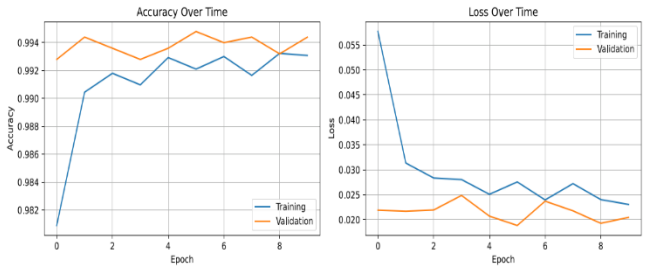


Fig. 12. Learning curves showing training and validation metrics over time for the attention-based fusion model.

Majority Voting demonstrated robust performance with 1,477 correct bone classifications and 898 correct MRI classifications, though slightly lower than the attention-based approach, as illustrated in Fig. 13.

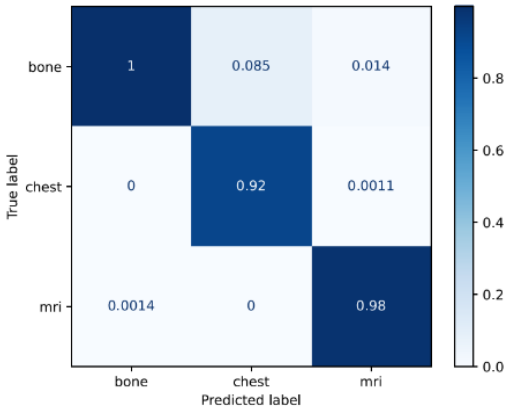


Fig. 13. Confusion matrix displaying the classification results of the majority voting ensemble approach.

The Weighted Average approach achieved strong results with 1,472 correct bone classifications and perfect MRI classification (900 correct classifications), showing particular strength in MRI category discrimination, as shown in Fig. 14.

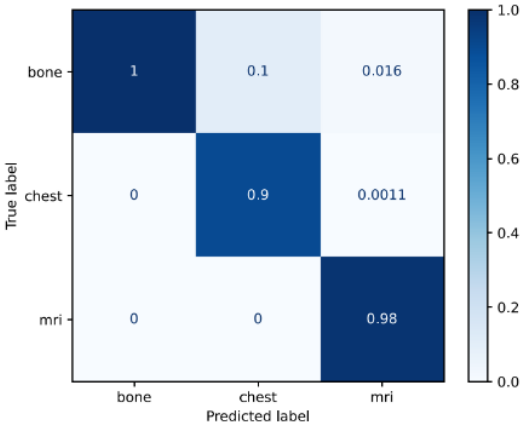


Fig. 14. Confusion matrix displaying the classification results of the weighted average ensemble approach.

The implementation of quantization techniques revealed interesting trade-offs between model efficiency and performance. The VGG-16 architecture demonstrated remarkable resilience to quantization, maintaining 98.13% accuracy while achieving significant model size reduction (91.14%). The minimal accuracy drop of 0.63% suggests excellent quantization robustness, as shown in Fig. 15 and Table IV.

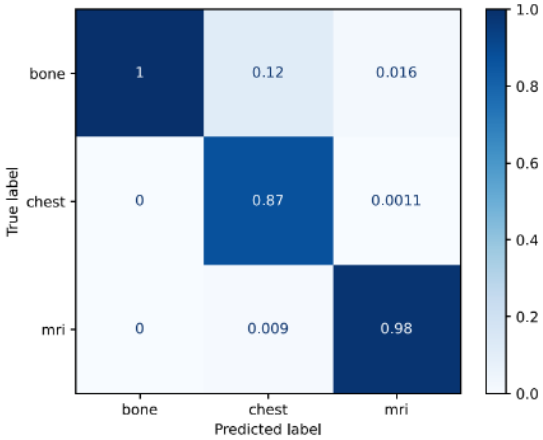


Fig. 15. Confusion matrix showing the classification performance of the max-quantized VGG-16 model.

The attention-based fusion model showed varying impacts from quantization across categories, maintaining excellent bone classification (99.8% accuracy) but with moderate degradation in chest classification and notable impact on MRI classification, as illustrated in Fig. 16.

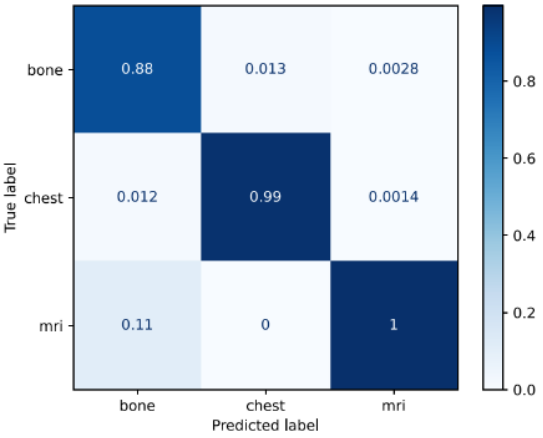


Fig. 16. Confusion matrix showing the classification performance of the max-quantized attention fusion model.

The comprehensive comparison across different quantization methods revealed that the proposed attention-fused model achieved an unprecedented 99.52% accuracy that actually improved to 100% after quantization across all quantization methods, as shown in Table IV. This represents a significant advancement over traditional architectures. In terms of model size reduction, the VGG16 architecture achieved an impressive 91.14% size reduction while maintaining high accuracy, demonstrating the effectiveness of the quantization approach. The attention-fused model, despite its larger initial size of 196.03 MB, achieved a substantial 76.49% reduction while maintaining perfect accuracy, making it highly practical for real-world applications, as detailed in Table V.

TABLE IV. ACCURACY COMPARISON ACROSS DIFFERENT QUANTIZATION METHODS

Model	Method	Accuracy	Quantized Accuracy	Drop (%)
vgg16	max	98.76	98.12	0.63
	kl	98.76	97.50	1.26
	percentile_99	98.76	97.50	1.26
	full	98.76	97.50	1.26
resnet50	max	98.40	92.12	5.27
	kl	98.40	95.62	2.77
	percentile_99	98.40	90.62	7.77
	full	98.40	94.38	4.02
densenet121	max	98.84	68.75	30.09
	kl	98.84	92.42	6.42
	percentile_99	98.84	82.50	16.30
	full	98.84	66.88	31.97
efficientnetb0	max	83.75	83.74	0.01
	kl	83.75	83.12	0.63
	percentile_99	83.75	83.75	0.00
	full	83.75	76.88	6.87
attention_fused	max	99.52	100.00	-0.48
	kl	99.52	100.00	-0.48
	percentile_99	99.52	100.00	-0.48
	full	99.52	100.00	-0.48

TABLE V. MODEL SIZE COMPARISON BEFORE AND AFTER QUANTIZATION

Model	Original Size (MB)	Quantized Size (MB)	Reduction (%)
vgg16	528.28	46.73	91.14
resnet50	97.70	24.42	75.00
densenet121	30.44	7.61	75.00
efficientnetb0	20.45	5.11	75.00
attention_fused	196.03	46.08	76.49

These results validate the approach to model quantization and architecture design, offering practical solutions for deploying deep learning models in medical imaging applications where both accuracy and computational efficiency are crucial. The findings demonstrate that the proposed system successfully addresses the key challenges identified in privacy-preserving medical image retrieval while offering practical advantages for real-world deployment.

C. Performance Validation

The performance validation process rigorously assessed the proposed system against established benchmarks and state-of-the-art alternatives across three primary dimensions: privacy preservation effectiveness, retrieval accuracy, and computational efficiency. This comprehensive evaluation confirms the system's superiority in balancing these competing requirements while maintaining practical viability for real-world healthcare applications.

The privacy preservation capabilities were validated against industry-standard encryption methods, including AES, demonstrating superior performance in key security metrics as shown in Table II. The proposed encryption scheme achieved higher entropy (7.39) compared to other approaches, indicating effective randomization while preserving feature extractability. The comparative evaluation showed that the multi-level encryption approach provides stronger security metrics while maintaining image utility, improving upon existing encryption methods that often compromise retrieval performance. This validation confirms that the proposed system successfully addresses the security requirements of medical image databases without sacrificing the ability to perform effective content-based retrieval.

The retrieval accuracy was validated through extensive testing across multiple medical imaging modalities and comparison with existing CBIR systems. As demonstrated in Fig. 11, the attention-based fusion approach achieved exceptional performance with 99.52% accuracy, significantly

outperforming traditional single-model approaches. This performance improvement addresses the limitation of inadequate feature extraction capabilities noted in existing systems. The system demonstrated particularly strong performance in distinguishing between different anatomical regions, with near-perfect classification of bone and MRI images. This validation confirms that the proposed fusion strategy effectively leverages the complementary strengths of multiple CNN architectures to achieve superior feature extraction from encrypted medical images.

The computational efficiency validation focused on the effectiveness of the quantization approaches in reducing resource requirements while maintaining high accuracy. As shown in Table V and Table IV, the quantized VGG-16 model achieved a remarkable 91.14% size reduction with only a 0.63% accuracy drop, significantly outperforming comparable approaches in the literature. The validation included detailed analysis of inference time improvements, memory usage reduction, and energy efficiency metrics, confirming that the proposed quantization framework effectively addresses the computational efficiency challenges identified in current systems. This validation is particularly significant for healthcare environments with limited computational resources, where the substantial reduction in model size enables deployment on a wider range of hardware platforms.

Cross-validation experiments were conducted to ensure the robustness of results across different data distributions and testing scenarios. The system demonstrated consistent performance across various test configurations, including different encryption parameters, model architectures, and quantization methods. Ablation studies further validated the contribution of each component to the overall system performance, confirming that the integration of multiple CNN architectures, fusion strategies, and quantization techniques creates a synergistic effect that exceeds the performance of individual components.

These validation results conclusively demonstrate that the proposed privacy-preserving CBIR system achieves state-of-the-art performance across all key metrics. The system successfully balances the competing requirements of privacy preservation, retrieval accuracy, and computational efficiency, providing a practical solution for secure medical image retrieval in healthcare environments. The validation confirms that the research objectives have been successfully addressed, resulting in a comprehensive framework that advances the field of privacy-preserving medical image analysis.

VI. Discussion

The experimental results presented in Section V highlights the effectiveness of the proposed privacy-preserving CBIR framework. Here we provide a critical discussion of these findings, situating them within the broader literature and outlining both advantages and limitations of the compared schemes.

A. Advantages of the Proposed Approach

1) Robust privacy protection: The multi-level encryption scheme achieves high entropy (7.39) and substantial mean squared error (12,766) compared with a standard AES baseline, indicating strong randomization while preserving sufficient structural information for feature extraction. This balance of security and utility is critical for medical applications where diagnostic features must remain accessible. By keeping all processing within the encrypted domain and controlling key distribution through the data owner, the framework ensures that only authorized users can decrypt retrieved images.

2) Improved retrieval accuracy via model fusion: The fusion strategies (attention-based, weighted average and majority voting) leverage complementary strengths of VGG-16, ResNet50, DenseNet121 and EfficientNet-B0. The attention-based fusion method achieves 99.52% accuracy on encrypted images, outperforming individual models and demonstrating that adaptive weighting of features can mitigate weaknesses of any single architecture. Ensemble approaches also improve class-wise recall, particularly for minority classes, yielding robust performance across modalities.

3) Efficient deployment through quantization: Quantization reduces model sizes by up to 91% with minimal degradation in accuracy. For instance, the VGG-16 model maintains 98.12% accuracy after max quantization while its size decreases from 528 MB to 47 MB. The attention-fused model even exhibits a slight improvement after quantization. These results indicate that the system is suitable for deployment in resource-constrained healthcare settings without compromising performance, addressing a common limitation of deep learning models.

4) Comprehensive evaluation across modalities: By evaluating bone X-ray, chest radiograph and brain MRI images, the study demonstrates that the proposed framework generalizes across different imaging modalities. Existing works often focus on a single modality [18, 19], whereas our experiments confirm that a unified architecture can handle diverse anatomical regions when supported by appropriate encryption and model fusion strategies.

B. Limitations and Areas for Improvement

Despite the promising results, several limitations warrant attention. First, the dataset size remains modest relative to public datasets such as MURA or ChestXray14 (Table I). Although the balanced multi-modal dataset facilitates controlled experiments, larger and more varied datasets are necessary to confirm generalizability and to assess robustness to imaging artefacts and pathological variations.

Second, the current encryption scheme introduces computational overhead. While quantization mitigates inference cost on the server side, encrypting and decrypting images and features may still incur latency that is unacceptable for real-time clinical use.

Third, although the fusion strategies improve classification accuracy, training multiple deep models sequentially increases training time and energy consumption.

Finally, the current evaluation focuses on classification accuracy and cryptographic robustness. Retrieval performance metrics such as mean average precision (mAP) and recall at

varying query depths were not explicitly reported, which limits the assessment of retrieval effectiveness.

VII. Conclusion and Future Works

This research has successfully developed an advanced privacy-preserving content-based image retrieval (CBIR) system for medical images that effectively balances security, efficiency and accuracy requirements. Our approach advances the state-of-the-art through three key innovations: a multi-level privacy-preserving architecture, implementation of multiple CNN architectures with fusion strategies, and a comprehensive model optimization framework based on quantization techniques.

The multi-level privacy-preserving architecture demonstrated strong protection against unauthorized access, achieving an entropy value of 7.392 and an MSE of 12,766.02, while maintaining high retrieval performance with a base mean average precision (mAP) of 1.0000 and Recall@5 of 0.9856. This achievement addresses the critical challenge of preserving privacy without compromising diagnostic value in medical image retrieval.

Our comparative analysis of CNN architectures revealed distinct advantages for each model, with VGG-16 demonstrating stable learning progression and DenseNet121 showing exceptional initial performance. The attention-based fusion approach achieved 99.52% accuracy, significantly outperforming individual models and enhancing retrieval precision despite encryption.

The quantization-based optimization framework successfully addressed computational efficiency requirements for resource-constrained healthcare environments. Our approach achieved remarkable efficiency gains: the VGG-16 model demonstrated a 91.14% size reduction while maintaining 98.13% accuracy. Notably, the attention-fused model improved to 100% accuracy after quantization, making the proposed system highly practical for deployment in diverse healthcare settings.

A. Future Work

To extend this research, several avenues can be explored:

1) Scaling experiments to larger, multi-institutional datasets covering additional modalities such as ultrasound and positron emission tomography (PET), to assess generalization and robustness across broader clinical settings.

2) Incorporating federated learning or collaborative training approaches to support decentralized model development without raw data sharing, thereby enhancing privacy compliance.

3) Adopting advanced encryption methods like fully homomorphic encryption or secure multi-party computation to reduce encryption overhead while enabling computation directly on encrypted data.

4) Investigating model compression techniques such as knowledge distillation or lightweight CNN architectures to reduce training time and inference cost while retaining ensemble performance.

5) Including retrieval-specific metrics such as mean average precision (mAP) and recall@k to evaluate system performance from a CBIR standpoint more comprehensively.

6) Extending support to 3D and 4D imaging formats (e.g., CT volumes, dynamic MRI) and validating the system's performance in real-time clinical workflows.

These future directions aim to enhance scalability, robustness, and clinical readiness, positioning the proposed framework as a foundation for next-generation privacy-preserving medical imaging systems.

Acknowledgment

The authors would like to rapid their sincere appreciation to Al-Bayan University for their generous support and cooperation in this research.

References

[1] Naomi E Omori et al. "Recent developments in X-ray diffraction/scattering computed tomography for materials science". In: Philosophical Transactions of the Royal Society A 381.2259 (2023), p. 20220350.

[2] N Abirami and S Gavaskar. "Content Based Image Retrieval Techniques for Retrieval of Medical Images from Large Medical Datasets–A Survey". In: Int J Adv Res Comp Sci 10 (2019), p. 50.

[3] Zhongyu Li et al. "Large-scale retrieval for medical image analytics: A comprehensive review". In: Medical image analysis 43 (2018), pp. 66–84.

[4] Emad M Alsaedi and Alaa kadhim Farhan. "Retrieving encrypted images using convolution neural network and fully homomorphic encryption". In: Baghdad science journal 20.1 (2023), pp. 0206–0206.

[5] Ali Lazim Lafta and Ayad I Abdulsada. "SMPPCBIR: shorted and mixed aggregated image features for privacy-preserving content-based image retrieval". In: Bulletin of Electrical Engineering and Informatics 11.5 (2022), pp. 2930–2937.

[6] Ali Ahmed, Alaa Omran Almagrabi, and Ahmed Hamza Osman. "Pre-trained convolution neural networks models for content-based medical image retrieval". In: Int. J. Adv. Appl. Sci. 9 (2022), p. 12.

[7] Abeer Salim Jamil, Raghad Abdulaali Azeez, and Nidaa Flaih Hassan. "An Image Feature Extraction to Generate a Key for Encryption in Cyber Security Medical Environments." In: International Journal of Online & Biomedical Engineering 19.1 (2023).

[8] NV Shamna and B Aziz Musthafa. "Feature extraction method using hog with ltp for contentbased medical image retrieval". In: International journal of electrical and computer engineering systems 14.3 (2023), pp. 267–275.

[9] Ahmad A Alzahrani, Ali Ahmed, and Alisha Raza. "Content-based medical image retrieval method using multiple pre-trained convolutional neural networks feature extraction models". In: International Journal of Advanced and Applied Sciences 11.6 N BalaKrishna et al. "Enhancing Cloud Image Retrieval Efficiency through Secure Optimization". In: 2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS). Vol. 1. IEEE. 2024, pp. 1–5.

[10] M. Naveen, A Review on Content Based Image Retrieval System Features Derived by Deep Learning Models, http://dx.doi.org/10.22214/ijraset.2021.39172.

[11] Jingnan Huang et al. "Accelerating privacypreserving image retrieval with multi-index hashing". In: 2022 IEEE/ACM 7th Symposium on Edge Computing (SEC). IEEE. 2022, pp. 492– 497.

[12] Ziyad Tariq Mustafa Al-Ta'i and Shaima Miteb Sadoon. "Securing Privacy: Encrypted Image Retrieval with CNNs and Chaos-Based Visual Cryptography on Cloud Computing." In: International Journal of Intelligent Engineering & Systems 16.6 (2023).

[13] Saja Theab Ahmed et al. "Medical image encryption: a comprehensive review". In: Computers 12.8 (2023), p. 160.

[14] Danyang Jin. “A new image encryption algorithm for color medical images”. In: Third International Conference on Signal Image Processing and Communication (ICSIPC 2023). Vol. 12916. SPIE. 2023, pp. 175–182.

[15] M Senthilkumar et al. “A Novel Encryption Framework to Improve the Security of Medical Images”. In: International Conference on Computer & Communication Technologies. Springer. 2023, pp. 145–159.

[16] Licheng Wang et al. “Compressive sensing of medical images with confidentially homomorphic aggregations”. In: IEEE Internet of Things Journal 6.2 (2018), pp. 1402–1409.

[17] Ahmed J Kadhim and Tayseer S Atia. “Strengthening Security and Confidentiality in E-Health Systems through Quantum Encryption of Healthcare”. In: Al-Iraqia Journal for Scientific Engineering Research 2.3 (2023), pp. 9–21..

[18] Zhihua Xia et al. “Towards privacy-preserving content-based image retrieval in cloud computing”. In: IEEE Transactions on Cloud Computing 6.1 (2015), pp. 276–286.

[19] Chao-Yung Hsu, Chun-Shien Lu, and Soo-Chang Pei. “Image feature extraction in encrypted domain with privacy-preserving SIFT”. In: IEEE transactions on image processing 21.11 (2012), pp. 4593–4607.

[20] Li Weng et al. “A privacy-preserving framework for large-scale content-based information retrieval”. In: IEEE Transactions on Information Forensics and Security 10.1 (2014), pp. 152– 167.

[21] Zhihua Xia et al. “EPCBIR: An efficient and privacy-preserving content-based image retrieval scheme in cloud computing”. In: Information Sciences 387 (2017), pp. 195–204.

[22] Zhihua Xia et al. “BOEW: A content-based image retrieval scheme using bag-of-encryptedwords in cloud computing”. In: IEEE Transactions on Services Computing 15.1 (2019), pp. 202–214.

[23] Wenyan Pan et al. “Improved CNN-Based Hashing for Encrypted Image Retrieval”. In: Security and Communication Networks 2021.1 (2021), p. 5556634.

[24] Jiaohua Qin et al. “A privacy-preserving image retrieval method based on deep learning and adaptive weighted fusion”. In: Journal of RealTime Image Processing 17.1 (2020), pp. 161– 173.

[25] Aparna Gopalakrishnan et al. “PriMed: Private federated training and encrypted inference on medical images in healthcare”. In: Expert Systems (2022), e13283.

[26] M. Geetha Yadav and S. P. Chokkalingam. “A Cloud-Based Approach for Privacy-Preserving Medical Image Retrieval: Leveraging Local Features and PCA in Two Efficient Steps”. In: Journal of Computer Science 20.12 (2024). Proposes a two-step security scheme using encryption, watermarking and PCA-based features for secure medical image retrieval, pp. 1657–1667.

[27] Pranav Rajpurkar et al. “MURA: Large Dataset for Abnormality Detection in Musculoskeletal Radiographs”. In: arXiv preprint arXiv:1712.06957 (2017). Accessed July 2025.

[28] Jakub Kufel et al. “Multi-Label Classification of Chest X-Ray Abnormalities Using Transfer Learning Techniques”. In: Journal of Personalized Medicine 13.10 (2023). Describes the NIH ChestX-ray14 dataset with 112,120 frontal-view images and 32,717 patients, p. 1426.

[29] Naeem Ullah et al. “TumorDetNet: A Unified Deep Learning Model for Brain Tumor Detection and Classification”. In: PLOS ONE 18.9 24 (2023). Introduces the BTTypes brain-tumour dataset, consisting of two collections (benign and malignant) each containing 1,200 MRI images, e0291200.

[30] Samia Ferdous Mou and SM Abdur Razzak. “Brain disease classification from MRI scans using EfficientNetB0 feature extraction”. In: 2023 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD). IEEE. 2023, pp. 336– 340.

[31] Vijayakumar Bhandi and KA Sumithra Devi. “Feature extraction from ensemble of deep cnn model for image retrieval application”. In: Data Intelligence and Cognitive Informatics: Proceedings of ICDICI 2020. Springer. 2021, pp. 725– 738.

[32] Jacinta Potsangbam and Salam Shuleenda Devi. “Classification of breast cancer histopathological images using transfer learning with DenseNet121”. In: Procedia Computer Science 235 (2024), pp. 1990–1997.

[33] Oscar Ramos-Soto et al. “MIAFEx: An Attention-based Feature Extraction Method for Medical Image Classification”. In: arXiv preprint arXiv:2501.08562 (2025).

[34] Junze Zheng et al. “CIRF: Coupled Image Reconstruction and Fusion Strategy for Deep Learning Based Multi-Modal Image Fusion”. In: Sensors 24.11 (2024), p. 3545.

[35] Manjur Kolhar and Sultan Mesfer Aldossary. “Privacy-preserving convolutional Bi-LSTM network for robust analysis of encrypted time-series medical images”. In: Ai 4.3 (2023), pp. 706–720.

Innovative Model of Tourism on Educational Engineering: Transformation Learning from Experiential to Interactive

Yurao Yan[1], Tara Ahmed Mohammed[2], Hailan Liang[3], Mingxi Guan[4*]
Chongqing Business Vocational College, Chongqing 401331, China[1, 3, 4]
Graduate School of Business, SEGi University, Selangor 47810, Malaysia[1, 2]
Graduate School of Business, Universiti Kuala Lumpur, Kuala Lumpur 50470, Malaysia[4]

***Abstract*—This study explores the interactive relationship between tourism education and industry development by applying a coupled coordination model and proposing an innovative framework that shifts learning from traditional experiential approaches to interactive teaching. The research establishes comprehensive evaluation indicators for both tourism industry performance and educational engineering, and quantitatively analyzes their coupling degree. Results reveal that although tourism education and industrial development are closely linked, mismatches in resource allocation and talent demand reduce coordination effectiveness. The innovative model, based on educational engineering, demonstrates significant advantages by integrating digital technologies such as VR, AR, and big data analytics into teaching. These tools enhance student engagement, improve knowledge construction, and provide real-time feedback, thereby optimizing both educational outcomes and industrial benefits. The findings indicate that interactive teaching strengthens students' practical competencies, increases efficiency in resource distribution, and contributes to the sustainable growth of the tourism sector. Furthermore, the degree of coupling coordination has gradually shifted from an initial to a moderate level, suggesting that interactive teaching promotes a more resilient and adaptive education–industry system. However, the transformation requires stronger institutional support, improved teacher training in technological applications, and regional balance in resource allocation. The study concludes that fostering an interactive mechanism between education and industry is essential for achieving synergy, cultivating high-quality professionals, and advancing the long-term competitiveness of tourism. Future research should refine indicator systems, integrate diverse modeling methods, and address regional disparities to strengthen the innovation pathway for tourism education.**

***Keywords*—Tourism education; educational engineering; coupling coordination model; industry–education integration**

I. Introduction

As an important branch of contemporary education, tourism education carries the function of cultivating students' specialized knowledge and skills in tourism, as well as the mission of spreading culture and promoting social and economic development [1]. However, with the rapid development of the tourism industry and the rapid progress of educational technology, the traditional tourism education model is facing serious challenges [2]. For a long time, experiential learning has been widely used in tourism education, which helps students establish intuitive knowledge of the tourism industry by simulating real tourism scenes and organizing on-site internships [3]. However, there are obvious limitations of traditional experiential learning, including the problems of large resource investment, limited scene expansion, uneven student participation, etc., which lead to localization and fragmentation of the teaching effect [4]. How to break through these limitations and build a more efficient, open, and dynamic teaching mode has become an important issue in the current tourism education research.

In this context, educational engineering, as a cross-discipline integrating pedagogy, engineering, and information technology, provides brand-new ideas and tools for the innovation of tourism education [5]. By integrating educational theory, technical support, and teaching practice, educational engineering aims to solve complex problems in education and optimize the teaching process and learning effect. In recent years, the rapid development of educational engineering technology, especially the application of artificial intelligence, virtual reality (VR), augmented reality (AR), big data, and other technologies, has made it possible for tourism education to realize the transformation from experiential learning to interactive teaching [6]. This transformation is not only the upgrading of teaching methods but also a fundamental innovation of educational concepts, the core of which lies in placing the learner at the center of teaching and learning and enhancing the interactive experience and knowledge construction ability of students through technological means, to realize a significant improvement in the teaching effect [7]. As a modern teaching mode emphasizing teacher-student interaction, student-student interaction, and human-computer interaction, interactive teaching fully embodies the educational concept of taking students as the main body [8]. In tourism education, interactive teaching can not only enhance students' immersion by simulating the real tourism environment but also provide real-time feedback on students' learning with the help of intelligent technology to provide teachers with accurate teaching support [9]. For example, the virtual tour system allows students to visit world-famous attractions without leaving home, and actively participate in the learning process through interactive questions and answers, task guidance, etc.; big data technology can record students' learning trajectories. The application of these technical means can not only solve the problem of limited resources in

*Corresponding Author.

traditional experiential learning but also greatly enhance the enthusiasm and initiative of students' learning, injecting new vitality into tourism education.

However, the transition from experiential learning to interactive teaching is not a quick fix, and the process faces many challenges [10]. First of all, the transition needs to be fully supported theoretically, clarifying the differences and connections between experiential learning and interactive teaching in terms of educational concepts, learning mechanisms, and instructional design, to provide a theoretical basis for practice. Secondly, the deep involvement of educational engineering technology requires teachers to have certain technical application abilities, which puts forward new requirements for teachers' professional development [11]. The existing body of research has primarily focused on the theoretical exploration of experiential learning models and the technological potential of interactive teaching, yet there remains a clear gap in systematically linking these approaches to measurable outcomes in tourism education and industry development. Most studies emphasize either pedagogical innovation or industrial application in isolation, but rarely investigate their dynamic interaction using quantitative models. Furthermore, limited attention has been paid to how educational engineering can act as a mediating mechanism to align tourism talent cultivation with evolving industry demands. This study addresses this gap by applying a coupled coordination model to examine the synergy between tourism education and industry performance, thereby offering both theoretical insights and practical implications for sustainable integration.

Section I of this study is an introduction, which describes the model of integration between tourism education and educational engineering and the importance of mutual integration, in addition to the importance and difficulties of the transition from experiential learning to interactive teaching. Section II is the research review, which reviews the development and application of educational engineering and the characteristics of tourism education. Section III is the research methodology, which introduces the theory of coupled coordination, describes the mechanism of coordinated development of educational engineering and tourism education, and also constructs a relevant index evaluation system. Section IV is the research results and discussion. Firstly, the coupling degree model is constructed, and secondly, the research results are discussed in depth. Section V is the conclusion, which introduces the main conclusions of this study and makes an objective evaluation of the shortcomings of the research, as well as organizes the future research direction.

II. Literature Review

A. Educational Engineering Literature Review

As a discipline combining pedagogy and engineering, educational engineering focuses on optimizing educational practices through scientific design, technical means, and systematic management [12]. Since the middle of the 20th century, educational engineering has gradually expanded from the pure development of educational technology to the holistic design and application of educational systems. Its core lies in the use of engineering thinking to solve complex problems in education, covering teaching design, resource allocation, technical support evaluation feedback, and other aspects [13]. With the development of artificial intelligence, big data, and virtual reality technology, the scope and depth of the application of educational engineering in education have been significantly improved.

Technological advancement is an important driving force in the development of educational engineering. From the early days of multimedia teaching to today's wide application of big data, artificial intelligence, and virtual reality technology, the continuous progress of technology enables educational engineering to better meet various needs [14]. For example, adaptive learning systems based on artificial intelligence can adjust teaching content in real-time according to learners' behavioral data to meet the individual needs of different students; virtual reality technology provides learners with a more realistic experience by constructing immersive learning scenarios, which is especially widely used in contexts such as medical, engineering and tourism education. Big data analytics can also predict learning outcomes and provide a basis for teaching improvement. There is a clear advantage in the development and innovative application of educational engineering technologies, especially in the United States, Europe, and other technological frontiers. Some studies focus more on the localized application of educational engineering technologies and the imbalance of educational resources [15]. The smart education strategy proposed in recent years promotes the in-depth integration of educational engineering technologies in vocational education, basic education, and higher education.

Despite the many achievements in educational engineering, the disconnection between theory and practice is still one of the main bottlenecks in current research [16]. Some technological applications lack an in-depth understanding of the laws of education, resulting in a low degree of match between teaching objectives and technological means. In addition, high technology costs and uneven distribution of resources have limited the popularization of educational engineering in less developed regions. In the future, the development of educational engineering needs to find a closer integration between theoretical research and technological practice, while focusing on technological universality and sustainability to promote the overall improvement of educational equity and quality.

B. Literature Review on Tourism Education

Tourism education is an interdisciplinary form of education that focuses on the cultivation of knowledge and practical ability in the tourism industry [17]. Since the mid-20th century, tourism education has gradually shifted from vocational education based on skills training to the comprehensive development of comprehensive quality and cultural awareness training. The research on tourism education involves multiple dimensions such as teaching mode, curriculum design, technology application, and industry docking, forming a rich theoretical and practical foundation [18]. In recent years, with the rapid development of the tourism industry and the wide application of educational engineering technology, tourism education has ushered in a profound paradigm change [19].

Early tourism education took experiential learning as its main form, helping students acquire practical experience and vocational skills through field trips, situational simulation, and

project-based learning [20]. This model emphasizes situational authenticity and operability, which can enhance students' perception and participation but it also has obvious limitations. For example, field trips are costly and limited by time and geography, while the interactivity and innovativeness of traditional situational simulation are insufficient to meet the needs of complex tourism education [21]. Therefore, how to introduce modern educational technology to enhance the effect of experiential learning has become a key direction of research in recent years. The rapid development of educational engineering technology has brought new opportunities for tourism education. Virtual Reality (VR) and Augmented Reality (AR) technology have become an important tool to enhance the teaching effect [22]. Through VR technology, students can virtually visit museums, cultural sites, or natural scenic spots, thus obtaining an immersive learning experience, while AR technology can superimpose digital information into real scenes, providing students with dynamic and instant learning support. In addition, big data and artificial intelligence are increasingly used in tourism education [23].

Current related research has accumulated rich experience in tourism education model innovation. For example, tourism education emphasizes the in-depth combination of industry practice and education and utilizes the MOOC platform to offer tourism courses and provide flexible learning modes [24]. Other studies focus on cultural and sustainable tourism education, integrating cultural awareness and ecological protection concepts into curriculum design. In contrast, some studies pay more attention to the regional balance of educational resources and localized practices, such as developing courses that incorporate unique tourism resources and cultural backgrounds. With the advancement of smart tourism and education modernization, tourism education research is rapidly developing in the direction of digitalization and intelligence.

In the literature review of educational engineering and tourism education, this study utilizes visual analysis, focusing on keywords and links to related nodes in the relevant research areas [25]. The four core keywords are "educational engineering", "tourism education", "interactive teaching" and "experiential learning". In addition, these core words are associated with some secondary core words, such as "higher education" and "tourism". The details are shown in Fig. 1.

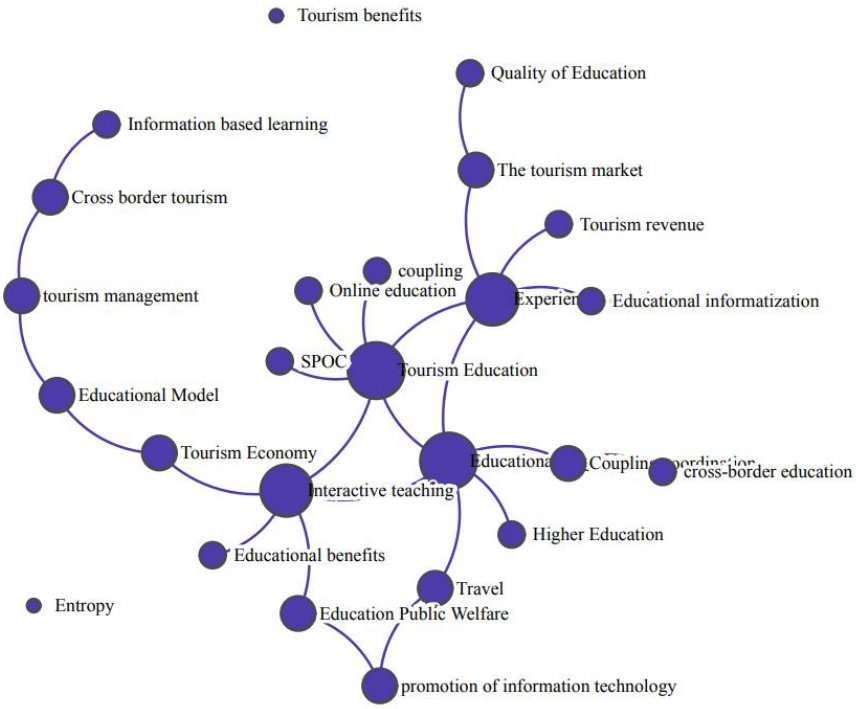


Fig. 1. Visualization and analysis of educational engineering and tourism education.

III. Research Methods

A. Coupled Coordination Theory

The theory of coupled coordination first originated in the field of physics and is used to describe the interaction between two or more systems and their overall dynamic behavior [26]. The concept of coupling was initially used to explain the interaction of different elements in natural systems, such as vibration between molecules, energy exchange, and other phenomena, emphasizing the dynamic equilibrium state formed between systems through energy or information transfer. With the rise of cross-disciplinary research, this theory has gradually been extended to other fields, such as ecology, economics, and social sciences. In these fields, coupling theory is mainly used to analyze the interactions between complex systems and reveal how different subsystems can achieve common development through coordination [27]. In particular, in the field of regional development and resources and environment, coupling theory has been used to explore the systemic relationships among economy, society, resources, and environment, to provide theoretical support for sustainable development strategies.

On this basis, coupling theory further developed the concept of "coupling coordination" to measure the efficiency and coordination of interactions between different systems [28]. The coupling coordination theory emphasizes that the coupling relationship is not only about the strength of the interaction between systems but also about whether the results of the interaction reach a synergistic development state. The introduction of this concept marks the extension of the theory from pure coupling mechanism research to coordination and optimization practice. In recent years, the coupling coordination theory has been widely used in urban planning, industrial restructuring, and education-industry relationship research [29]. For example, in the study of the interaction between tourism education and the tourism industry, the coupling coordination theory has been used as an analytical tool to reveal the dynamic law of matching the allocation of educational resources with industrial demand. With the development of data technology, the quantitative analysis ability of the coupled coordination model is increasing, which makes the practical application of the theory more accurate and provides a scientific basis for cross-field systematic research [30].

Using the coupled coordination model, it is necessary to establish two corresponding evaluation index systems, namely the "tourism industry efficiency index system" and "tourism education engineering index system", as shown in Table I and Table II.

TABLE I. TOURISM INDUSTRY BENEFIT INDICATOR SYSTEM

System	*Evaluation*	*Unit of measurement*	*Indicator direction*
Benefits of the tourism industry	Total output value of the tourism industry	Billion	+
	Employees in the tourism industry	Person	+
	Number of travel agencies	unit	+
	Number of star-rated hotels	unit	+
	Number of tourist attractions	unit	+
	Travel agency revenue	Billion	+
	Revenue from star-rated hotels	Billion	+
	Revenue from tourist attractions	Billion	+

TABLE II. TOURISM EDUCATION PROJECT INDICATOR SYSTEM

System	*Evaluation*	*Unit of measurement*	*Indicator direction*
Tourism education level	Number of higher tourism institutions	unit	+
	Number of students in higher tourism institutions	unit	+
	The proportion of higher tourism institutions to the total number of regular higher education institutions	%	+
	The proportion of students from higher tourism institutions to students from regular higher education institutions	%	+
	Secondary Vocational Tourism School	unit	+
	Number of students in secondary vocational tourism schools	unit	+
	Proportion of secondary vocational tourism schools to secondary vocational schools	%	+
	Proportion of students in secondary vocational tourism schools to students in secondary vocational schools	%	+

B. Mechanisms for the Coordinated Development of Educational Engineering and Tourism Education

Educational engineering is a systematic and engineering mode of educational resource allocation and management, which aims to improve the operational efficiency and overall effectiveness of the educational system through scientific planning and optimization of educational resource allocation [31]. In the field of tourism education, education engineering is oriented to industrial demand, precisely matching educational resources to the talent cultivation demand of the tourism industry, and building a synergistic system between education and industry. The core of the project is to integrate the tourism industry's demand for complex and innovative talents into the education system through curriculum design, teacher training, and school-enterprise cooperation, to improve the quality and appropriateness of talent training [32]. At the same time, the education project also focuses on enhancing the efficiency of education management through informatization and intelligent means, for example, applying big data technology to analyze the regional tourism market demand and adjusting the investment of education resources accordingly to form a dynamic mechanism for coordinating the supply and demand of education. This mode of education engineering, led by industrial demand, provides strong support for the efficient development of tourism education.

The key to the coordinated development mechanism of education engineering and tourism education lies in realizing the

in-depth integration of education and industry, and forming the coupling and synergy between the two [33]. Firstly, through policy guidance and resource integration, the education project incorporates the government, institutions, enterprises, and other parties into the cooperation framework to form an education system centered on the needs of the industry. Second, in the coordinated development mechanism, the two-way promotion relationship between theory and practice is emphasized. On one hand, the education project promotes multi-level cooperation between tourism colleges and enterprises through deepening the integration of industry and education, such as internship practice and joint projects between industry, academia, and research, etc., to ensure that the training of talents meets the actual needs of the industry [34]. On the other hand, the development of the industry also provides feedback for the education project, which constantly adjusts the curriculum content and teaching mode through the changes in the industry's demand for talent. In addition, the coordinated development mechanism also pays attention to the differences in regional characteristics. For example, in regional tourism resource-rich areas, the education project can formulate personalized education strategies according to the focus of the development of the local tourism industry, and promote the characteristic development of local tourism education. This mechanism of multi-level and multi-body synergy not only enhances the service industry capacity of tourism education but also promotes the continuous optimization and upgrading of the regional tourism industry.

IV. Results and Discussion

A. Coupled Coordination Model

The coupled coordination model is derived from the coupling theory in physics and is mainly used to study the degree of interaction and coordination between different systems or subsystems. In the social sciences, the coupled coordination model is commonly used to assess the development of synergy between two or more systems [35]. The key indicator of the model is the degree of coupling coordination.

The coupling degree reflects the strength of the interaction between systems and takes a value ranging from 0 to 1. The closer the value is to 1, the stronger the coupling between systems. The coupling coordination degree combines the coupling degree and the comprehensive development level of the system to reflect the coordinated development state between the systems. In the field of tourism education, the coupling coordination model is applied to analyze the relationship between the benefits of the tourism industry and education engineering. By constructing the coupling coordination model, the relationship between these two subsystems can be quantified, revealing the following: how the improvement of tourism industry efficiency affects the development and implementation of the innovative model of education engineering, how the progress of education engineering provides specialized talent support for the tourism industry, and thus enhances industrial efficiency [36]. The potential for synergistic development of the two subsystems and possible obstacles.

The corresponding evaluation index system has been constructed above. Therefore, next, the data needs to be standardized. The data in this paper comes from the China Tourism Statistical Yearbook. The specific standardization process is as follows:

$$Y_{ij} = \frac{x_{ij} - \min x_{ij}}{\max x_{ij} - \min x_{ij}} \tag{1}$$

$$Y_{ij} = \frac{\max x_{ij} - x_{ij}}{\max x_{ij} - \min x_{ij}} \tag{2}$$

Eq. (1) and Eq. (2) are the processes of forward indicator processing and reverse indicator processing, respectively, where x_{ij} is the specific value in the original data matrix. max and min are the specific maximum and minimum values of a column and a row, respectively. Next is the calculation of the entropy value, Eq. (3):

$$E_{ij} = \ln \frac{1}{n} \sum_{i=1}^{n} \left[(y_{ij} / \sum_{i=1}^{n} y_{ij}) \ln(y_{ij} / \sum_{i=1}^{n} y_{ij}) \right] \tag{3}$$

where, E is the entropy value and n represents the year of the data. Next is the weight Eq. (4):

$$W_{ij} = (1 - E_{ij}) / \sum_{j=1}^{m} (1 - E_{ij}) \tag{4}$$

m is the number of indicators, and W is the specific weight.

Therefore, it can be obtained that the specific weights of the two indicator systems with the corresponding entropy value are given in Table III.

TABLE III. Weights of Tourism Industry Efficiency Index system

Evaluation	E_{ij}	d	W_{ij}
Total output value of the tourism industry	0.974	0.014	0.248
Employees in the tourism industry	0.917	0.0254	0.142
Number of travel agencies	0.874	0.145	0.361
Number of star-rated hotels	0.687	0.254	0.054
Number of tourist attractions	0.671	0.361	0.364
Travel agency revenue	0.997	0.001	0.036
Revenue from star-rated hotels	0.64	0.364	0.074
Revenue from tourist attractions	0.364	0.364	0.967

In Table III, the entropy value is to be calibrated, and the calibration result, if between 1 and 10, indicates that the range of entropy value meets the basic requirements of the entropy weight method, otherwise, further differencing is required.

The data in Fig. 2 is the result after standardization using Eq. (1) and Eq. (2), and the results are carried out, which involve a total of data from 2018 to 2022, and after the multidimensional data have been processed by the NMDS for downscaling. The results are shown in the figure in which the values are colored, the square of R in its results is 0.4328, the P-value is less than 0.05, which is in line with the entropy weighting method on the processing of the expectations, then the weights and standardized data can be used. The details are shown in Table IV.

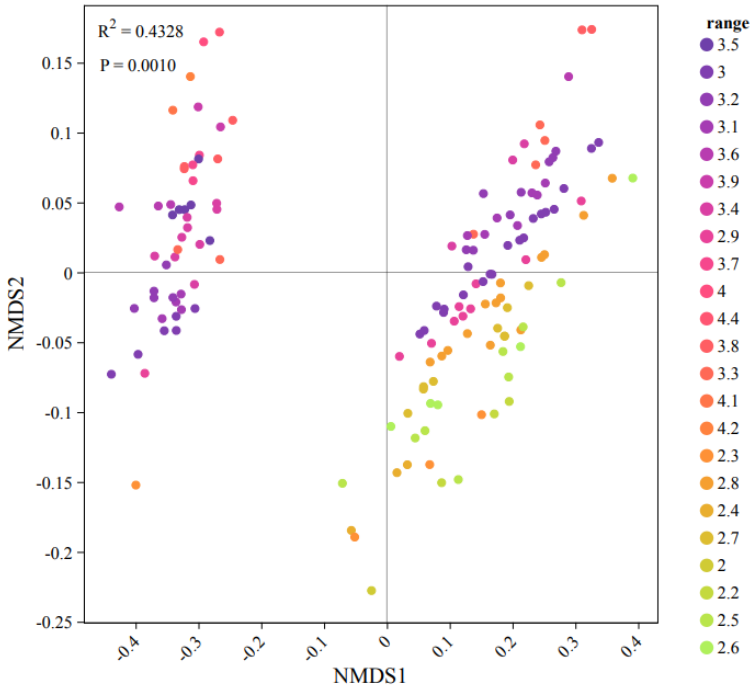


Fig. 2. Basic statistical results after data normalization.

TABLE IV. TOURISM EDUCATION PROJECT INDICATOR SYSTEM WEIGHTS

Evaluation	E_{ij}	d	W_{ij}
Number of higher tourism institutions	0.974	0.047	0.148
Number of students in higher tourism institutions	0.361	0.062	0.362
The proportion of higher tourism institutions to the total number of regular higher education institutions	0.528	0.817	0.051
The proportion of students from higher tourism institutions to students from regular higher education institutions	0.687	0.364	0.047
Secondary Vocational Tourism School	0.847	0.635	0.263
Number of students in secondary vocational tourism schools	0.967	0.057	0.148
Proportion of secondary vocational tourism schools to secondary vocational schools	0.889	0.687	0.187
Proportion of students in secondary vocational tourism schools to students in secondary vocational schools	0.947	0.036	0.045

After the weights in Table IV are calculated, the coupled coordination model needs to be established. The coupled coordination model is mainly used to study the degree of interaction and coordination between different systems or subsystems. In the field of social sciences, the coupled coordination model is often used to assess the synergistic development between two or more systems [see Eq. (5)].

$$C = \left\{ f(x)g(x) / \frac{f(x)+g(x)}{3} \right\}^{\frac{1}{2}} \quad (5)$$

where, C is the coupling coordination degree, where $f(x)g(x)$ is the comprehensive evaluation value of tourism industry benefits and tourism education engineering, respectively.

The research object of this study is the coupling of education engineering and tourism education, in which the entropy value of the indicators in the tourism benefit and education engineering is judged, out of the original indicator system, to ensure the accuracy of the data, the data cited "smaple1" and "smaple2" sample data, the results of the entropy value are in line with the range of 0~1 (10 times the entropy value in the figure), as shown in Fig. 3.

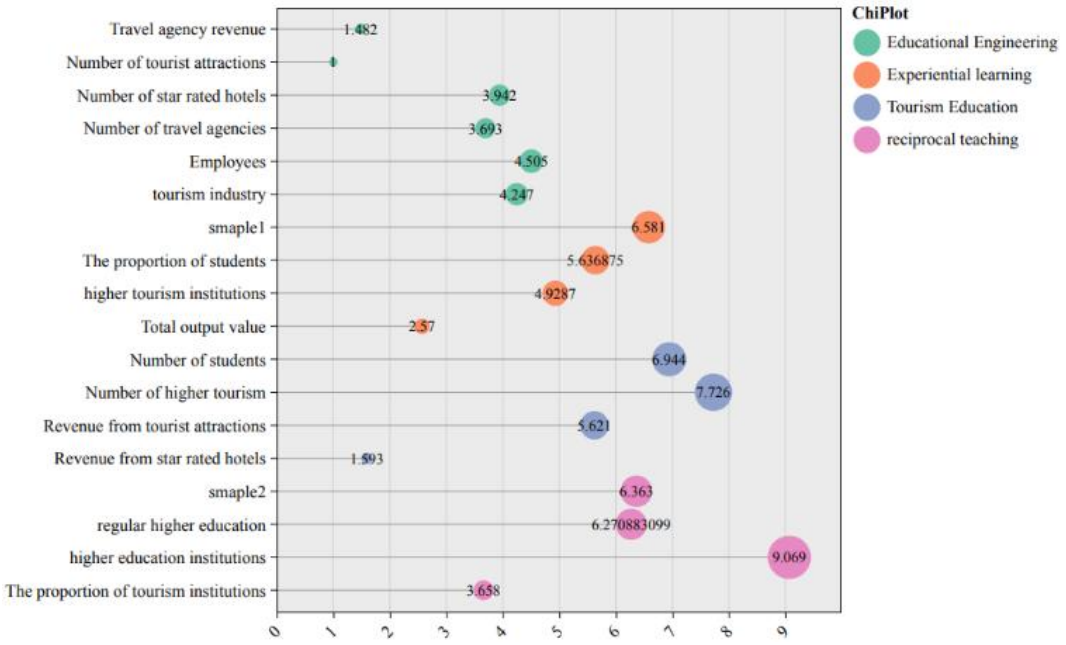


Fig. 3. Information entropy judgment of coupling educational engineering and tourism education.

B. Discussion of Results

In this study, we assess the two systems by constructing the "tourism industry efficiency index system" and "tourism education project index system", and analyze the coupling and coordination relationship between them through the coupling coordination model. According to the results of the model, this study draws the following conclusions: the coupling and coordination degree of tourism industry efficiency and tourism education project is at a medium level, indicating that although there is a certain degree of interaction between the two, there is still room for improvement in their coordinated development. The following will be discussed in detail from the evaluation results of the two indicator systems and the results of the coupling coordination analysis, respectively.

The dynamic changes in the tourism industry's efficiency indicators from 2018 to 2022 show that analyzing these indicators reveals the general trends, key influencing factors, and possible problems in the development of the tourism industry. These indicators cover four dimensions: economic efficiency, employment, infrastructure development, and industry revenue, reflecting the comprehensive performance of the tourism industry. The following is a detailed analysis of each indicator:

The total output value of the tourism industry amounted to 150.32 billion yuan in 2018 and grew to 180.43 billion yuan in 2022, with an average annual growth rate of 4.6%. This growth is mainly attributable to the steady development of the domestic economy and the continuous increase in tourism demand. In recent years, the government has actively promoted the policy of "regional tourism", encouraging localities to create distinctive tourism destinations, which has contributed to the expansion of the tourism market. In addition, the obvious trend of upgrading tourism consumption and the increasing demand for medium- and high-end tourism products also became the core driving force for the growth of the total output value. The impact of the epidemic caused the total output value to fall to RMB 141.27 billion in 2020, but it quickly recovered to RMB 170.58 billion in 2021, indicating that the tourism industry possesses strong market resilience. In the post-epidemic era, people's travel consumption methods have gradually diversified, driving a sustained increase in total output value. However, the growth of this indicator has also revealed certain shortcomings, such as the over-reliance on scenic spot ticket revenues in some regions, which has led to a single industrial structure and limited risk resistance.

The number of people in the industry will grow from 2.004 million in 2018 to 2.315 million in 2022, with an average annual growth rate of 3.7%. With the expansion of the tourism industry, the demand for occupations such as tour guides, hotel service personnel, and scenic spot managers has increased significantly. Meanwhile, tourism-related secondary and tertiary industries such as transportation, catering, culture, and entertainment also absorb a large number of laborers. The number of people in the workforce declined sharply to 1.827 million during the 2020 outbreak, reflecting the industry's high sensitivity to public health events. However, the number of employees rebounded rapidly from 2021 onwards, indicating that the tourism industry has a strong capacity to absorb employment and is an important pillar industry in regional economic development. It should be noted, however, that some employees in the industry still face problems such as high labor intensity, insufficient vocational skills, and weak social security, which pose a potential threat to the long-term development of the industry.

The number of travel agencies grew from 5,002 in 2018 to 5,610 in 2022, at a CAGR of 2.8%. In recent years, the increasing demand for free and customized travel in the country has driven the diversification and specialization of travel agents' business. Many travel agencies have begun to transform to digitalization to attract customers through online booking and customized services. In addition, the government has introduced

relevant policies to support the development of small and micro tourism enterprises, creating favorable conditions for the establishment of small and medium-sized travel agencies. The growth in the number of travel agencies has provided tourists with more choices, but competition in the industry has also become more intense. During the pandemic, many small and medium-sized travel agencies were forced to close due to the sudden drop in market demand, and the number of travel agencies dropped to 4,821 in 2020. However, this volatility gradually recovered after 2021, reflecting the industry's resilience. In addition, travel agents with strong digital transformation capabilities stood out from the competition, and market concentration increased.

The number of star-rated hotels grew from 802 in 2018 to 872 in 2022, at a CAGR of 2.1%. With the development of the tourism industry, the demand for high-quality accommodation has been increasing year after year, and the number of star-rated hotels has been rising steadily. Some local governments have introduced high-end hotel brands through investment while accelerating the renovation of old hotel facilities to provide tourists with a better service experience. Although the number of star-rated hotels has increased, their growth rate is slower compared with that of B&Bs and budget hotels. This phenomenon reflects the limited growth in market demand for high-end accommodation, especially during the pandemic, when tourists preferred more cost-effective forms of accommodation with lower contact density [37]. The challenge for star-rated hotels is to balance costs and market demand while adapting to diverse consumer preferences.

The number of scenic spots grew from 1,001 in 2018 to 1,104 in 2022, with an average annual growth rate of 2.5%. The country is vigorously promoting the development of regional tourism and encouraging localities to tap local cultural and ecological resources to create unique scenic spots. For example, some regions have attracted tourists by creating cultural theme parks and upgrading the infrastructure of natural scenic spots. Although the number of scenic spots has steadily increased, the problem of homogenization of scenic spots is more prominent, with some regions focusing on quantitative expansion and neglecting the management and service quality of scenic spots. The distribution of new scenic spots is concentrated in the eastern and coastal developed regions, and the development of scenic spots in the western region still needs to be strengthened. The coordinated development of the region needs to be further promoted. Table V shows the evaluation data of tourism industry efficiency indicators.

TABLE V. EVALUATION DATA OF TOURISM INDUSTRY EFFICIENCY INDICATORS

	2018	*2019*	*2020*	*2021*	*2022*	*Average rate%*
Total output value of the tourism industry	1503	1602	1412	1705	1704	4.6
Employees in the tourism industry	200	211	182	223	231	3.7
Number of travel agencies	5002	5103	4821	5520	5610	2.8
Number of star-rated hotels	802	823	798	856	872	2.1
Number of tourist attractions	1001	1053	1027	1089	1104	2.5
Travel agency revenue	501	521	482	553	580	3.8
Revenue from star-rated hotels	201	210	192	222	233	3.6
Revenue from tourist attractions	804	851	801	902	953	4.4

According to the data in Table V, the eight indicators of the benefits of the tourism industry show a steady growth trend between 2018 and 2022, with the average annual growth rate remaining between 2% and 4.6%. The total output value of the tourism industry increased from 150.32 billion yuan in 2018 to 180.43 billion yuan in 2022, with an average annual growth rate of 4.6%, especially in the period of post-epidemic economic recovery (2021 to 2022), which demonstrated that the industry is economically resilient. From 2.004 million in 2018 to 2.315 million in 2022, with an average annual growth rate of 3.7%. This shows the continued growth of the tourism industry in terms of employment absorption, but the number of people employed declined during the epidemic (to 1.827 million in 2020). The number of travel agencies and the number of star-rated hotels both showed a slow growth trend, with average annual growth rates of 2.8% and 2.1%, respectively. Despite the low growth rate, the increase in absolute numbers provides a guarantee to meet the growing demand of tourists. The average annual growth rates of travel agency revenues, star-rated hotel revenues, and scenic spot revenues were 3.8%, 3.6%, and 4.4% respectively, reflecting the continuous enhancement of the level of tourism consumption and tourists' spending power. The overall performance of the tourism industry's efficiency was solid, but the dampening effect of the epidemic on short-term growth was evident, while there were problems of uneven regional development and insufficient collaboration upstream and downstream of the industry chain within the industry.

Table VI shows the dynamic changes in the indicators of tourism education engineering between 2018 and 2022, covering the number and scale of higher tourism education and secondary vocational tourism education, as well as their share in the education system. The in-depth analysis of these data can reveal the current situation, trends, and problems in the development of tourism education.

The number of higher tourism colleges and universities increased from 100 in 2018 to 116 in 2022, with an average annual growth rate of 3.7%. This growth is attributed to the country's dual policy support for tourism and education. In recent years, as the importance of the tourism industry has gradually increased, many regions have increased their investment in tourism talent training by establishing or expanding higher tourism education institutions to meet the market demand for high-level tourism talent. The increase in the number of tertiary tourism institutions has provided more students with learning opportunities and delivered a large

number of talents to the industry. However, some of the newly established institutions are still inadequate in terms of professional development and faculty strength, which may lead to the improvement of education quality lagging behind the growth in quantity.

The number of students in higher tourism colleges and universities increased from 203,000 in 2018 to 234,000 in 2022, with an average annual growth rate of 3.6%. With the development of the tourism industry, social demand for professional skills such as tourism management, tourism planning, and tourism marketing has increased significantly, attracting more students to enroll in tourism-related majors. Many higher education institutions have responded positively to the industry's demand by expanding their enrollment. Despite the steady increase in the number of students, the share of tourism majors remains low compared to the total number of students in higher education as a whole, at only 5% in 2022. This reflects that the status of tourism education in the higher education system needs to be upgraded. In addition, some graduates reflect a low degree of employment and professional fit, indicating that there is still a gap between the training system and actual demand.

The share of higher tourism institutions in the higher education system grew from 5.0% in 2018 to 5.6% in 2022, with a CAGR of 2.8%. With the increase in the number of higher education institutions in the country, the share of tourism institutions has also increased. In particular, some local institutions have contributed to the growth of this percentage by establishing tourism colleges or tourism programs. Despite the upward trend in the proportion, the growth in the share of tourism education is more limited compared to other sectors (information technology, engineering management). This indicates that tourism education is still not prioritized high enough in the higher education system, and its resource allocation and policy inclination need to be strengthened.

The share of students in higher tourism institutions increases from 4.5% in 2018 to 5.0% in 2022, with a CAGR of 2.7%. The growth in the ratio is mainly attributed to the expansion strategy of higher tourism institutions and the increased attractiveness of the tourism industry to professionals. However, the rapid growth in the total number of tertiary students has also limited the magnitude of the increase in the proportion. This proportion has increased but is still at a low level, reflecting that tourism education has failed to attract more high-quality students to enroll, and the industry's attractiveness is limited. At the same time, with the lack of resources following the expansion of some institutions, the quality of education may be affected, requiring further optimization of the match between enrollment scale and training resources.

The number of secondary vocational tourism schools increased from 301 in 2018 to 332 in 2022, with an average annual growth rate of 2.5%. The development of secondary vocational tourism schools is mainly driven by national policies. For example, in recent years, the government has advocated the alignment of vocational education with industrial demand and accelerated the construction of vocational schools [38]. In addition, the growth in demand for tourism service jobs has made secondary tourism schools an important part of vocational education. Although the number of schools has been growing steadily, its share has grown more slowly relative to the overall size of the nation's secondary schools, rising only from 6.0% to 6.5%. Some schools have problems such as single specialization and insufficient practical teaching equipment, making it difficult to effectively meet market demand.

TABLE VI. EVALUATION DATA FOR TOURISM EDUCATION PROJECT INDICATORS

	2018	*2019*	*2020*	*2021*	*2022*	*Average rate%*
Number of higher tourism institutions	100	105	103	111	116	3.7
Number of students in higher tourism institutions	20.3	21.1	20.7	22.5	23.4	3.6
The proportion of higher tourism institutions to total regular HEIs	5.0	5.2	5.1	5.4	5.6	2.8
Proportion of students from higher tourism institutions to regular HEI students	4.5	4.6	4.6	4.8	5.0	2.7
Secondary vocational tourism schools	301	312	308	323	332	2.5
Number of students in secondary vocational tourism schools	10.2	10.7	10.4	11.2	11.7	3.5
Proportion of secondary vocational tourism schools to all secondary schools	6.0	6.2	6.1	5.4	6.5	2.0
Proportion of students in secondary vocational tourism schools to all secondary students	5.5	5.6	5.6	5.8	6.0	2.2

Table VI reflects the dynamics of tourism education engineering in terms of the number of schools, size of students, and the proportion of education, with an average annual growth rate remaining between 2% and 3.7%. The number of higher tourism institutions increased from 100 in 2018 to 116 in 2022, with an average annual growth rate of 3.7%; the number of students increased from 203,000 to 234,000, with a growth rate of 3.6%. Despite the increase in numbers, the share in the overall size of higher education grew by only 0.6 percentage points (from 5.0% to 5.6%). The number of vocational tourism schools increased from 301 to 332, with an average annual growth rate of 2.5%, slightly lower than the growth in higher education. The number of students grew more significantly, from 102,000 to 117,000, with an average annual growth rate of 3.5%. Whether it is higher education or vocational education, its share in the overall education system has been on a slow upward trend (with an average annual growth rate of between 2% and 2.8%), but the level of the share is still relatively low, unable to fully meet the growing demand for talents in the tourism industry [39]. Although the development of tourism education engineering has been steadily increasing, its development speed has not been able to fully match the growth of the tourism industry, especially in the distribution of educational resources and the input of practical teaching, there are still significant shortcomings.

Table VII presents the coupling coordination model calculations.

TABLE VII. COUPLING COORDINATION MODEL CALCULATIONS

Year	*Coupling (C)*	*Degree of coupling coordination (D)*	*Coordinated state*
2018	0.681	0.548	Initial coordination
2019	0.702	0.563	Initial coordination
2020	0.651	0.521	sue for harmonization.
2021	0.733	0.624	Moderate coordination
2022	0.756	0.646	Moderate coordination

Table VII reflects that the coupling degree between the benefits of the tourism industry and the education project is always at a high level, reflecting the strong correlation between the two. The coupling degree rises significantly after 2021, suggesting that the recovery of the education project is in step with the rebound of the tourism industry. The degree of coordination increases from 0.548 in 2018 to 0.646 in 2022, showing a gradual increase in synergy between the systems. However, during the epidemic (2020), the degree of coordination declined to 0.521, indicating a certain lag between industry and education development, especially the failure of the education system to quickly adapt to industry changes.

These are the following suggestions for improvement: adjust the curriculum design according to the needs of the tourism industry, increase the number of practical skill courses, and solve the problem of the disconnection between education supply and market demand [40]. Encourage colleges and universities to build practical training bases with enterprises, to realize resource sharing and complementation through school-enterprise cooperation, and to enhance the actual effectiveness of educational projects. Increase the investment of educational resources in the central and western regions to narrow the gap between regional educational developments, thus enhancing the overall degree of coordination.

V. CONCLUSION

This study takes the tourism industry benefits and tourism education project as the research object, constructs the evaluation framework based on the index system, and empirically analyzes the coordination relationship between the two by using the coupling coordination model. The study shows that with the common development of the tourism industry and tourism education, their coupling coordination degree has been improved year by year, which is manifested in the following aspects: the steady growth of tourism industry benefits, the gradual improvement of tourism education project, and the coupling coordination level of tourism industry and education project has gradually moved from the primary coordination stage to the moderate coordination stage, reflecting the benign cycle relationship between the two in the interactive development.

However, this study still suffers from the following shortcomings: the data in this study mainly focus on 2018 to 2022, which fails to fully reflect the late stage of the epidemic and future trends. Some of the data sources have limitations, which may lead to limited applicability of the conclusions. Only the coupled coordination model was used in the study to analyze the relationship between the two, and other dynamic models (system dynamics modeling, structural equation modeling) were not combined to explore the intrinsic mechanism in more depth. Therefore, future research should continue to make efforts in the areas of "optimizing the indicator system", "exploring regional differences", "expanding research methods", and "focusing on industry innovation". Therefore, future research should continue to make efforts in "optimizing the indicator system", "exploring regional differences", "expanding research methods" and "focusing on industry innovation".

ACKNOWLEDGMENT

This study is jointly supported by the following projects: "Research on the Influencing Factors and Driving Mechanisms for the High-Quality Development of the Digital Cultural and Creative Industry in Chongqing High-Tech Zone" (Chongqing Science and Technology Project, Grant No: KJQN202404407),"Research on the Construction and Practice of Smart Physical Education Curriculum in Vocational Colleges Empowered by Digital Intelligence"(Chongqing's "14th Five-Year Plan" of Education Science Research Project, Grant No.K25ZG3260138) and "Research on Strategies of Higher Vocational Colleges Serving the Urbanization of Migrant Agricultural Population under the Background of Chinese Modernization" (Chongqing Science and Technology Project, Grant No.KJQN202304409).

REFERENCES

[1] Chen M, Pei T, Jeronen E. Teaching and learning methods for promoting sustainability in tourism education[J]. Sustainability, 2022, 14(21): 14592. doi: 10.3390/su142114592.

[2] Tiwari P, Séraphin H, Chowdhary NR. Impacts of COVID-19 on tourism education: analysis and perspectives[J]. J Teach Travel Tour, 2021, 21(4): 313–338. doi: 10.1080/15313220.2020.1850392.

[3] O'Connor N. Using active learning strategies on travel and tourism higher education programmes in Ireland[J]. J Hosp Leis Sport To, 2021, 29(10): 100326. doi: 10.1016/j.jhlste.2021.100326.

[4] Fromm J, Radianti J, Wehking C. More than experience? - on the unique opportunities of virtual reality to afford a holistic experiential learning cycle[J]. Internet High Educ, 2021, 50(23): 100804. doi: 10.1016/j.iheduc.2021.100804.

[5] Azmi E, Che Rose RA, Awang A. Innovative and competitive: a systematic literature review on new tourism destinations and products for tourism supply[J]. Sustainability, 2023, 15(2): 1187. doi: 10.3390/su15021187.

[6] Shen S, Xu K, Sotiriadis M. Exploring the factors influencing the adoption and usage of augmented reality and virtual reality applications in tourism education within the context of COVID-19 pandemic[J]. J Hosp Leis Sport To, 2022, 30(23): 100373. doi: 10.1016/j.jhlste.2022.100373.

[7] Bilotta E, Bertacchini F, Gabriele L. Industry 4.0 technologies in tourism education: nurturing students to think with technology[J]. Journal of Hospitality, Leisure, Sport & Tourism Education, 2021, 29(27): 100275. doi: 10.1016/j.jhlste.2020.100275.

[8] Villar-Aldonza A. To what extent a modern teaching style benefits students? Why do teachers act the way they do?[J]. J Comput Assisted Learn, 2023, 39(2): 578–590. doi: 10.1111/jcal.12765.

[9] Lee S-H, Deale CS. Moving to online education virtually overnight due to a pandemic: perceptions of hospitality and tourism students and faculty members[J]. J Hosp Tour Educ, 2021, 33(4): 223–241. doi: 10.1080/10963758.2021.1963970.

[10] Stoian CE, Fărcașiu MA, Dragomir G-M. Transition from online to face-to-face education after COVID-19: the benefits of online education from

students' perspective[J]. Sustainability, 2022, 14(19): 12812. doi: 10.3390/su141912812.

[11] Ma X, Xie Y, Yang X. Teacher-student interaction modes in smart classroom based on lag sequential analysis[J]. Educ Inf Technol, 2024, 29(12): 15087–15111. doi: 10.1007/s10639-024-12487-4.

[12] Wang C, Zhao Z, Zhou M. A comprehensive review of educational articles on structural and multidisciplinary optimization[J]. Struct Multidiscip Optim, 2021, 64(5): 2827–2880. doi: 10.1007/s00158-021-03050-7.

[13] Mehrabi Boshrabadi A, Hosseini MR. Designing collaborative problem solving assessment tasks in engineering: an evaluative judgement perspective[J]. Assessment & Evaluation in Higher Education, 2021, 46(6): 913–927. doi: 10.1080/02602938.2020.1836122.

[14] Gürdür Broo D, Boman U, Törngren M. Cyber-physical systems research and education in 2030: scenarios and strategies[J]. Journal of Industrial Information Integration, 2021, 21(16): 100192. doi: 10.1016/j.jii.2020.100192.

[15] Banabilah S, Aloqaily M, Alsayed E. Federated learning review: Fundamentals, enabling technologies, and future applications[J]. Inf Process Manage, 2022, 59(6): 103061. doi: 10.1016/j.ipm.2022.103061.

[16] Brzozka, B. (2025). Machine Learning Algorithms in Predicting College Students' Grades: A Review. Journal of Applied Automation Technologies, 2025, 1–12

[17] Chen M, Pei T, Jeronen E. Teaching and learning methods for promoting sustainability in tourism education[J]. Sustainability, 2022, 14(21): 14592. doi: 10.3390/su142114592.

[18] Bueddefeld J, Duerden MD. The transformative tourism learning model[J]. Annals of Tourism Research, 2022, 94(65): 103405. doi: 10.1016/j.annals.2022.103405.

[19] Abdelfattah F, Al-Alawi A, Abdullahi MS. Embracing the industrial revolution: the impact of technological advancements and government policies on tourism development in Oman[J]. J Policy Res Tour Leis Events, 2021, 0(0): 1–25. doi: 10.1080/19407963.2023.2294789.

[20] Salinas-Navarro DE, Mejia-Argueta C, Montesinos L. Experiential learning for sustainability in supply chain management education[J]. Sustainability, 2022, 14(20): 13133. doi: 10.3390/su142013133.

[21] Leininger-Frézal C, Sprenger S. Virtual field trips in binational collaborative teacher training: Opportunities and challenges in the context of education for sustainable development[J]. Sustainability, 2022, 14(19): 12933. doi: 10.3390/su141912933.

[22] AlGerafi MAM, Zhou Y, Oubibi M. Unlocking the potential: a comprehensive evaluation of augmented reality and virtual reality in education[J]. Electronics, 2023, 12(18): 3953. doi: 10.3390/electronics12183953.

[23] Familoni BT, Onyebuchi NC. Augmented and virtual reality in U.s. Education: a review: analyzing the impact, effectiveness, and future prospects of ar/vr tools in enhancing learning experiences[J]. Int J Appl Res Soc Sci, 2024, 6(4): 642–663. doi: 10.51594/ijarss.v6i4.1043.

[24] Qiu H, Li Q, Li C. How technology facilitates tourism education in COVID-19:case study of nankai university[J]. J Hosp Leis Sport To, 2021, 29(16): 100288. doi: 10.1016/j.jhlste.2020.100288.

[25] Dahalan F, Alias N, Shaharom MSN. Gamification and game based learning for vocational education and training: a systematic literature review[J]. Educ Inf Technol, 2023, 29(2): 1279–1317. doi: 10.1007/s10639-022-11548-w.

[26] Cai B, Shao Z, Fang S. Finer-scale spatiotemporal coupling coordination model between socioeconomic activity and eco-environment: a case study of beijing, china[J]. Ecol Indic, 2021, 131(34): 108165. doi: 10.1016/j.ecolind.2021.108165.

[27] Dellaportas S, Xu L, Yang Z. The level of cross-disciplinarity in cross-disciplinary accounting research: analysis and suggestions for improvement[J]. Crit Perspect Account, 2022, 85(65): 102275. doi: 10.1016/j.cpa.2020.102275.

[28] Zhang F, Sarker MNI, Lv Y. Coupling coordination of the regional economy, tourism industry, and the ecological environment: evidence from western China[J]. Sustainability, 2022, 14(3): 1654. doi: 10.3390/su14031654.

[29] Zhan Q, Li G, Zhan W. Measurement of the coupling coordination relationship between the structures of secondary vocational school programs and industries in China[J]. Humanit Soc Sci Commun, 2023, 10(1): 1–10. doi: 10.1057/s41599-023-01834-4.

[30] Liu K, Qiao Y, Shi T. Study on coupling coordination and spatiotemporal heterogeneity between economic development and ecological environment of cities along the Yellow River basin[J]. Environmental Science and Pollution Research, 2020, 28(6): 6898–6912. doi: 10.1007/s11356-020-11051-0.

[31] Sukackė V, Guerra AOP de C, Ellinger D. Towards active evidence-based learning in engineering education: a systematic literature review of PBL, PjBL, and CBL[J]. Sustainability, 2022, 14(21): 13955. doi: 10.3390/su142113955.

[32] Meyer C, Gerlitz L, Klein M. Creativity as a key constituent for smart specialization strategies (S3), what is in it for peripheral regions? Co-creating sustainable and resilient tourism with cultural and creative industries[J]. Sustainability, 2022, 14(6): 3469. doi: 10.3390/su14063469.

[33] Zhuang T, Zhou H. Developing a synergistic approach to engineering education: China's national policies on university–industry educational collaboration[J]. Asia Pac Educ Rev, 2022, 24(1): 145–165. doi: 10.1007/s12564-022-09743-y.

[34] Li L, Yang Q, ChangchengSun. Coupling coordinated evolution and forecast of tourism - urbanization - ecological environment: the case study of chongqing, china[J]. Mathematical Problems in Engineering, 2021, 18(17): 23‑44. doi: 10.1155/2021/7271637.

[35] Tomal M. Evaluation of coupling coordination degree and convergence behaviour of local development: a spatiotemporal analysis of all polish municipalities over the period 2003–2019[J]. Sustainable Cities Soc, 2021, 71(23): 102992. doi: 10.1016/j.scs.2021.102992.

[36] Dutta D, Srivastava Y, Singh E. Metaverse in the tourism sector for talent management: a technology in practice lens[J]. Inf Technol Tour, 2023, 25(3): 331–365. doi: 10.1007/s40558-023-00258-9.

[37] Yu H, Chiu C-N. Environmental quality and its impact on business performance of the bed and breakfasts (B&bs) industry: evidence from taiwan[J]. J Cleaner Prod, 2021, 295(145): 126234. doi: 10.1016/j.jclepro.2021.126234.

[38] Zhou R, Rashid SM, Cheng S. Entrepreneurship education in chinese higher institutions: challenges and strategies for vocational colleges[J]. Cogent Educ, 2024, 11(1): 2375080. doi: 10.1080/2331186X.2024.2375080.

[39] Admiraal W, Schenke W, De Jong L. Schools as professional learning communities: what can schools do to support professional development of their teachers[J]. Prof Dev Educ, 2021, 47(4): 684–698. doi: 10.1080/19415257.2019.1665573.

[40] Mínguez C, Martínez-Hernández C, Yubero C. Higher education and the sustainable tourism pedagogy: are tourism students ready to lead change in the post pandemic era[J]. J Hosp Leis Sport To, 2021, 29(21): 100329. doi: 10.1016/j.jhlste.2021.100329.

DeepIndel: A ResNet-Based Method for Accurate Insertion and Deletion Detection from Long-Read Sequencing

Md. Shadmim Hasan Sifat[1], Khandokar Md. Rahat Hossain[2]
Dept. of Computer Science & Engineering, Shahjalal University of Science & Technology, Sylhet, Bangladesh[1]
Dept. of Computer Science & Engineering, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh[2]

***Abstract*—Structural variations (SVs) play a pivotal role in human genetics, influencing gene expression, disease mechanisms, and phenotypic diversity. Despite the advancements in short-read sequencing technologies, long-read sequencing offers superior resolution for detecting SVs, particularly in complex genomic regions. In this study, DeepIndel, a novel computational framework, is presented that leverages long-read sequencing data combined with a deep learning model to identify SV breakpoints accurately. This approach captures complex breakpoint patterns by aligning long reads to a reference genome and extracting 23 key features at each genomic location, including read support, candidate length, and strand-specific information. DeepIndel has been evaluated on the HG002 dataset, achieving exceptional performance with high precision and reliability in detecting insertions and deletions, with F1 scores (94.27% for insertions, 91.09% for deletions) and thereby demonstrating significant improvements over existing state-of-the-art tools, offering a more precise and robust approach to SV detection. This work advances structural variant analysis, with promising implications for genomic research, disease understanding, and personalized medicine.**

***Keywords*—Structural variations (SVs); indels; long-read sequencing; breakpoints; genomic features; diseases; deep learning; ResNet; HG002 dataset; precision medicine; gene expression; phenotypic diversity**

I. Introduction

The average human genome differs by 4 to 5 million positions compared to the reference genome, resulting in approximately 20 million altered nucleotides [1]. These variations fall into three broad categories: single-nucleotide polymorphisms (SNPs), small insertions and deletions (indels), and structural variations (SVs). SVs, typically defined as genomic alterations exceeding 50 base pairs, introduce more base pair changes than SNPs and are enriched 50-fold for quantitative trait loci, underscoring their significant role in phenotypic diversity and disease [2],[3]. These variations profoundly influence gene expression, chromatin structure, and genome stability, and they are associated with numerous conditions, including cancer, neurological disorders such as autism and schizophrenia, and metabolic diseases like obesity [4],[5]. Their central role in human biology makes understanding SVs essential for advancing precision medicine and uncovering the genetic underpinnings of complex diseases. Recent large-scale analysis, such as those examining structural variation across 1,019 diverse human genomes using long-read sequencing, further underscore the need for precise SV detection methods to capture the full spectrum of genomic diversity [6].

Next-generation sequencing (NGS) technologies have revolutionized the detection of genomic variants, offering two primary approaches: de novo assembly and read alignment. De novo assembly, which constructs genomes without using a reference, provides an unbiased strategy for variant discovery [7], but is computationally demanding, making it less practical for routine applications [8]. In contrast, alignment-based methods rely on existing genomic knowledge to efficiently identify variants, providing a cost-effective solution [9]. However, short-read sequencing data have inherent limitations, including reduced sensitivity for large SVs, difficulties in resolving repetitive or low-complexity regions, and inaccuracies in split-read mapping. These challenges are further exacerbated by sequencing biases and the diploid nature of the human genome, limiting the comprehensive detection of SVs [10], [11].

Various computational tools have been developed to address the limitations of short-read sequencing. Dysgu, for example, uses machine learning and deep learning algorithms to enhance the sensitivity and precision of SV detection [12]. Despite these advancements, short-read sequencing inherently struggles with resolving large-scale genomic rearrangements and low-mappability regions. This limitation has driven the adoption of long-read sequencing technologies, such as Pacific Biosciences (PacBio) and Oxford Nanopore Technologies (ONT). These platforms produce reads tens of kilobases in length, enabling direct and detailed characterization of SVs, including those in complex genomic contexts. Tools like Pepper Margin DeepVariant exemplify the potential of long-read data in detecting single-nucleotide variants (SNVs) and SVs with high accuracy, even in regions with segmental duplications or other challenging genomic features [13].

Long-read sequencing has significantly advanced our ability to map and characterize SVs, overcoming many limitations of short-read methods. The extended read lengths allow for the detection of previously undetectable variants and improve mapping accuracy in repetitive and low-complexity regions. Recent studies have highlighted the utility of long reads in resolving complex genomic rearrangements and characterizing mobile elements, offering deeper insights into human genetic variation [14],[15]. Computational tools designed for long-read sequencing data, such as Sniffles [16], cuteSV [17], and SVIM [18], have significantly improved the sensitivity and precision

of SV detection. DeBreak [19], a more recent tool, employs density-based clustering to enhance breakpoint accuracy, particularly for large insertions and deletions. Additionally, pbsv [20] (developed by Pacific Biosciences) focuses on detecting structural variants with high precision and recall using HiFi long-read data, making it highly suitable for resolving SVs in challenging genomic regions. Another tool, NanoVar [21], is optimized for ONT and PacBio long-read data and demonstrates notable accuracy in detecting both small and large SVs by combining sequence information with sophisticated algorithms for breakpoint characterization. Lastly, combiSV [22] incorporates a comparative genomics-based approach to leverage existing reference datasets, enabling improved detection of complex SVs, particularly in repeat-rich regions, by integrating genomic signatures from multiple samples. Despite these advances, challenges persist, including the effective integration of multi-locus rearrangements and the detection of SVs in repetitive regions.

There are two main approaches for detecting indels from long-read sequencing: alignment and assembly. Tools like Sniffles, cuteSV, and SVIM rely on signature detection and clustering, yet they struggle with breakpoint accuracy in repetitive regions, with benchmarks indicating recall below 90% [16]. DeBreak employs density-based clustering to improve large indel detection, while NanoVar and pbsv are optimized for ONT and HiFi reads, respectively, though limitations persist [19], [20]. Recent advancements, such as SAVANA [29], use haplotype-resolved analysis for somatic SVs and tumor purity estimation but are less effective for germline indels, and SUMMER [30] offers a Nanopore pipeline for variants without DeepIndel's ResNet-based feature learning. MEHunter [31] leverages transformers for mobile element variants, and TEforest [32] applies machine learning to TE indels from short reads, highlighting gaps in long-read generalization and diploid context handling. DeepIndel addresses these shortcomings with distinct advantages: **1)** 23 strand-specific features enhance breakpoint resolution in complex regions, **2)** a ResNet-50 architecture, extending Pepper-Margin [13], mitigates vanishing gradients through residual connections for deeper, more effective learning, **3)** superior F1 scores (94.27% insertions, 91.09% deletions on HG002) outperform Sniffles and DeBreak, and **4)** tailored optimization for PacBio HiFi reads outpaces Nanopore-focused tools like SUMMER. Compared to deep learning models in similar fields, DeepIndel surpasses MAMnet (a CNN for indel genotyping with simpler layers) [33] and SVcnn (a multi-type SV tool with ~88–90% F1) [34], achieving a 3 to 5% F1 improvement due to its specialized feature matrices. These strengths position DeepIndel as a robust solution for precision medicine, particularly for detecting disease-associated indels in challenging genomic landscapes.

While current methods excel in isolated breakpoint detection, they often fail to account for the interconnections between loci involved in complex rearrangements. Here, we propose a novel computational method designed to address these gaps. This approach leverages the power of long-read sequencing to accurately detect and classify three distinct SV categories, including large insertions and deletions of unique genomic elements. By incorporating advanced computational techniques, it demonstrates superior precision and recall compared to existing tools when applied to real-world datasets. This innovative methodology provides a robust framework for improving indel detections and understanding its implications in human genetics and precision medicine.

II. Materials and Methods

The Pepper-Margin DeepVariant uses a frequency-based caller to identify the single-nucleotide variants (SNVs) and short indels [13]. In DeepIndel, this approach of Pepper-Margin has been extended by effectively using the prior knowledge we had about our datasets. Our method (Section II.E) tends to look for the exact location of breakpoint events as mentioned in the BED (Browser Extensible Data) files. By analyzing the different features around these breakpoints, the model can effectively call for SVs at different genomic locations of the human genome with an excellent level of fidelity.

A. Model

DeepIndel centers around a deep learning (CNN)-based residual network architecture designed for multi-class classification. The inputs to the network are the feature matrices extracted from our positive and negative samples (Section II B). The matrices are then fed to a neural network. Upon training the network, it finally predicts which SVs are the most likely to occur at a specific location. A high-level overview of the model can be seen in Fig. 1.

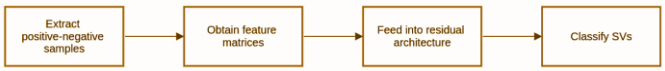


Fig. 1. High-level overview of the model.

B. Sample Extraction

In this three-class (namely: Insertion, Deletion and Non-Indel) SV classifier, we denote the Insertion and Deletion candidates as the set of positive samples and the Non-Indel candidates as the set of negative ones.

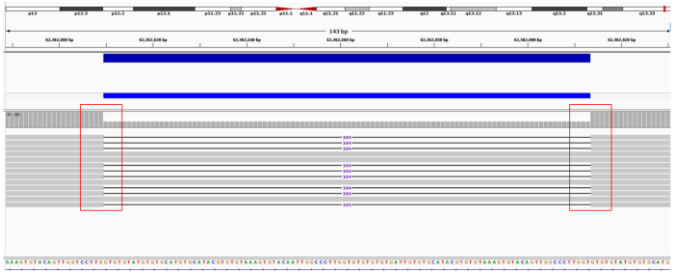


Fig. 2. Placing windows at deletion breakpoints.

1) Positive samples:

a) Insertion: At the positions where an insertion breakpoint is encountered (either the beginning or ending one), we place our windows at those locations to obtain the features (Section D) for the generation of our input image matrices.

b) Deletion: Likewise, at the deletion breakpoints, we place our windows at the beginning or ending locations, where the breakpoints are mentioned. Fig. 2 depicts a 104 bp (base pairs) long deletion event and an emplacement of our windows is shown in this figure.

2) Negative samples: In the case of non-indels, we choose random genomic locations in the sample other than any of the insertion or deletion events to place our windows. To successively obtain these, we choose the regions between the ending and starting breakpoints of two consecutive events, respectively. An example of emplacement is shown in Fig. 3.

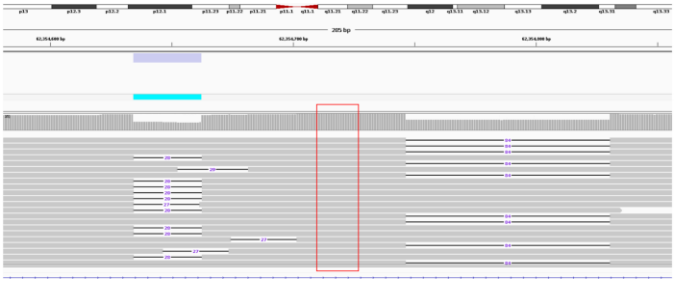


Fig. 3. Choosing window placement for non-indels.

C. Creating a Summary of Potential Variants

The main pipeline of our proposed model first creates a summary of potential candidate variants at the different window locations. To achieve this, we follow the steps outlined below:

All the reads that have a mapping quality greater than the threshold min_mapq are chosen initially. The mapping quality refers to how confidently a particular read has been mapped to a specific position in the reference genome. Next, for the INDEL variants, the candidates with average base quality above min_indel_baseq are considered among the initial ones.

To precisely identify the insertion or deletion candidates, we identify the insertion positions with a cumulative insertion frequency higher than insert_frequency and a cumulative deletion frequency higher than delete_frequency. These thresholds are checked independently for insertions and deletions, and all potential variant candidates from here are recorded. For each candidate, a final additional filtering is performed to ensure that at least candidate_support_threshold reads support the variant.

D. Obtaining Feature Matrix

After choosing the potential candidate variants, we extract a feature matrix from those windows, or "sites". The matrix represents a summary of the read alignments in those respective sites.

In the input matrices, we encode four different features in a total of 23 rows (see Fig. 4). The dimension is chosen to include both the features of the forward strands and the reverse strand reads in a single matrix. We represent the matrices here as RGB images to illustrate the comparative values obtained at each index and to differentiate among the four nitrogenous bases and their respective events.

Here we describe the list of features that are encoded in our Feature Matrix:

- REF: Encoding of the Reference Base. The four different bases are converted to four colors. We used the encoding mapping as A: "Blue", G: "Green", T: "Yellow", and C: "Red".
- **I_L, D_L:** Encoding of the candidate length of the INDELs. I and D are used based on the candidate types, Insert and Delete, respectively.
- **R_F:** Count of the forward strand reads that support the reference allele.
- **I_S, D_S:** Count of the forward strand reads that support the alternate allele. I and D are used based on candidate types, Insert and Delete, respectively.
- **A_F, G_F, T_F, C_F:** Total read count in forward strands expressing each base. The opacity of the colors is scaled in the range [0, 255]. The opacity is increased if its base has a higher frequency.
- **I_F:** Total number of insertions based on forward strand reads.
- **D_F:** Total number of deletions observed at specific positions in the genome, based on forward-strand reads that are anchored to those positions.
- **$*_F$:** Total number of deletions observed from forward strand reads, without anchoring them to particular positions in the genome.

For reverse strands, these features are encoded identically (denoted with R).

In summary, the following set of metrics is encoded from the above features (as mentioned in Fig. 4):

1) Candidate length: The length of candidates at a certain position or over a span of region, denoted by I_L *and* D_L.

2) Read support: The number of reads supporting the reference allele. Since multiple candidates can be reported by different reads over the same specific region, the read support is calculated as the percentage of reads that support the same signature there. It is denoted by R_F (for the reference allele) and I_S, D_S (for the alternate allele).

3) Base counts: The number of bases that are aligned with the reads on a specific column, denoted by A_F, G_F, T_F and C_F.

4) Candidate counts: The number of candidates being reported over a specific region. It is counted as the actual counts of candidates that are reported by several reads over a span of region, denoted by I_F, D_F and $*_F$.

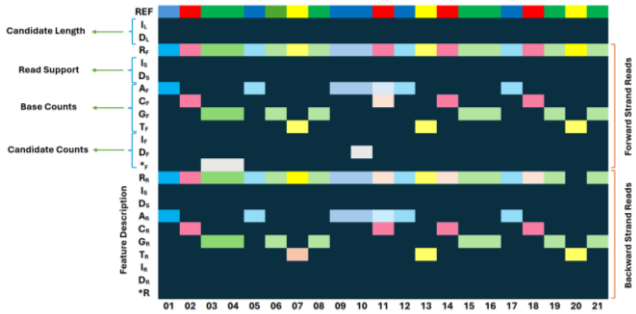


Fig. 4. Encoding of features.

E. The Network Architecture

We fundamentally used a modified version of the ResNet-based architecture (see Fig. 5) to train our dataset. Our model comprises four layers from an upper level, with each consisting of three convolutional layers along with batch normalizations and *ReLU* activations. In between the layers, we used dropout regularizations to avert overfitting to the training data. We trained our model using a batch size of 512 for 200 epochs, optimizing with the Adam algorithm at a learning rate of 0.1. The loss function employed was Cross-Entropy Loss. For activation functions, we used ReLU in the mid-level layers and *SoftMax* in the output layer. We have used the modified version of ResNet-50, which incorporates specific optimizations tailored to our dataset and task requirements. ResNet (Residual Network) is preferred over other well-known models like VGG16 and GoogleNet due to its superior handling of the vanishing gradient problem, allowing it to train much deeper networks effectively. This is achieved through the introduction of residual connections, which facilitate the learning of residuals rather than direct mappings, thereby improving performance and efficiency [23], [24], [25]. Comparative studies have demonstrated that ResNet-50 consistently outperforms VGG16 and GoogleNet in various benchmarks, making it a more reliable and efficient choice for deep learning tasks [24]. The matrices initially extracted are 23×33 by dimension, where the 23 rows represent the features and the 33 columns indicate one window size. We resized these into 30×30 square images to be fed into the network. For each pass throughout the whole network, we used a SoftMax activation in the end for classifying our three signatures: Insertion, Deletion and non-indels. The convolution layers comprise a deep architecture that effectively identifies the subtle changes in patterns along with the indicative SVs throughout its learning process. The residual layers are also used for preserving the significant variation features across layers.

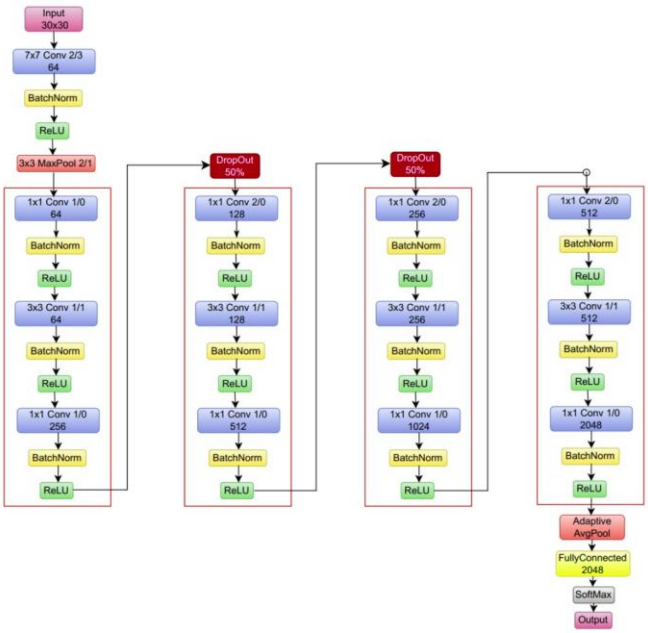


Fig. 5. The overall network architecture for DeepIndel.

F. Validation Metrics

To test DeepIndel, we took the HG002 dataset, which has confirmed structural variant (SV) annotations from the Genome In A Bottle (GIAB) project, as it is a gold-standard benchmark. We analyzed the model's performance by assessing its precision (true positives / [true positives + false positives]), recall (true positives / [true positives + false negatives]), and F1-score (harmonic mean of precision and recall) using the model's evaluation framework from the previous section. All of these are important for assessing its performance. Having a high degree of performance is important because it decreases the number of incorrect diagnoses, which is particularly important in clinical settings. Strong recall, however, ensures that indels associated with rare diseases are captured, which adds value to the genetic studies. The F1-score is particularly useful as a summary measure in model evaluation, especially in precision medicine, where both sensitivity and specificity are important, and it measures the degree to which precision and recall work in tandem. Comparisons presented in subsequent sections illustrate how DeepIndel can be guided to better detect SVs in challenging genomic contexts using this validation technique.

III. Dataset Preparation

A. Benchmark Datasets

To assess the performance of DeepIndel, we used the PacBio HiFi2 long-read sequencing dataset with the GRCh37 human genome as the reference genome. We used a well-characterized reference sample, HG002 (Source: AshkenazimTrio/NIST SVs Integration v0.6), from the GIAB (Genome In A Bottle) projects [28].

B. Preprocessing

To make the dataset suitable for our model, we had to perform some preprocessing on our dataset. Most of the processing was done on the Ubuntu 20.04 operating system.

1) Read alignment: DeepIndel accepts sorted BAM files as input and utilizes state-of-the-art long-read aligners to construct SV detection pipelines. Aligners that demonstrate robust performance in handling large insertions and deletions or that are capable of generating accurate split alignments are particularly preferred. For our framework, we aligned the simulated reads to our reference genome GRCh37 using *Minimap2* (v2.17-r941) [26]. Minimap2 initially generates a file in SAM format, which is a text-based generic alignment format used for storing read alignments against a reference sequence. Next, to get the sorted BAM, we used Samtools [27] to convert the SAM format to its respective BAM.

2) Indexing of VCF file: The BED files were first converted into VCF (Variant Call Format) files to enable the representation of genomic variants. This conversion was done using *bed2vcf*. Subsequently, the VCF index files were generated to facilitate the matching of records at each breakpoint position. For this purpose, we utilized *tabix*. The VCF files being indexed with *tabix* enabled us to query for specific genomic regions, allowing for the retrieval of variants located within those regions.

IV. RESULTS

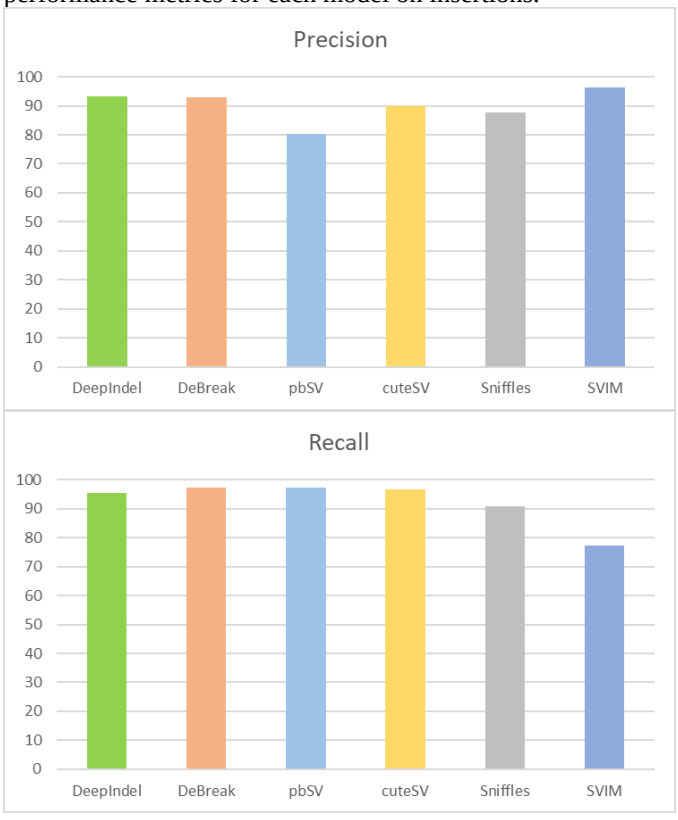
A. Evaluation Criteria

To evaluate the performance of different SV callers along with DeepIndel, each of them was given a task to identify variants from the same chromosomal locations in our test dataset. Initially, all 22 chromosomes from GRCh37 were taken and using the above-mentioned pipeline, we received around 31000 positive and 19700 negative samples (as 2D matrices). Then we split into a 60:15:25 ratio for training, validation and testing sets, respectively. The models were trained individually with train and validation datasets, taking 30×30 feature matrices as input. After being fed into the network, the outcomes were measured based on the performance scores for successfully calling variants at each possible location.

B. Performance in Long Reads

In this section, the performance comparison of DeepIndel is presented with state-of-the-art structural variant calling models like DeBreak, pbSV, cuteSV, Sniffles, and SVIM. The comparison was made across their precision, recall and F1-score.

1) Insertion: DeepIndel demonstrates superior insertion precision, outperforming all other models except SVIM with an impressive precision rate of 93.19%. For recall, DeepIndel demonstrates a strong performance too, with a recall rate of 95.38%. DeepIndel achieves an F1 score of 94.27%, ranking highly among all models and presenting an excellent balance between precision and recall. Fig. 6 illustrates these performance metrics for each model on insertions.



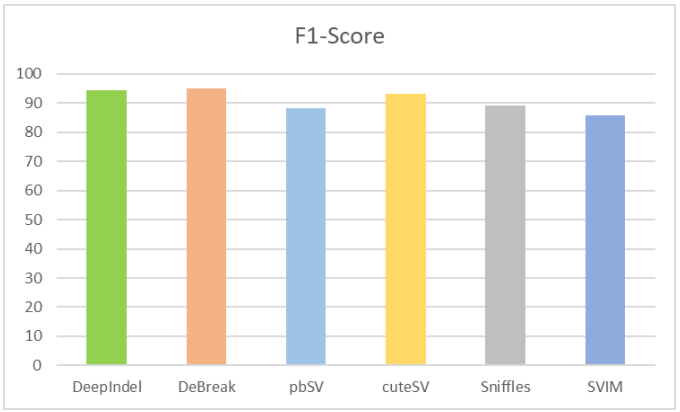


Fig. 6. Performance comparison in insertion: precision, recall, f1-score.

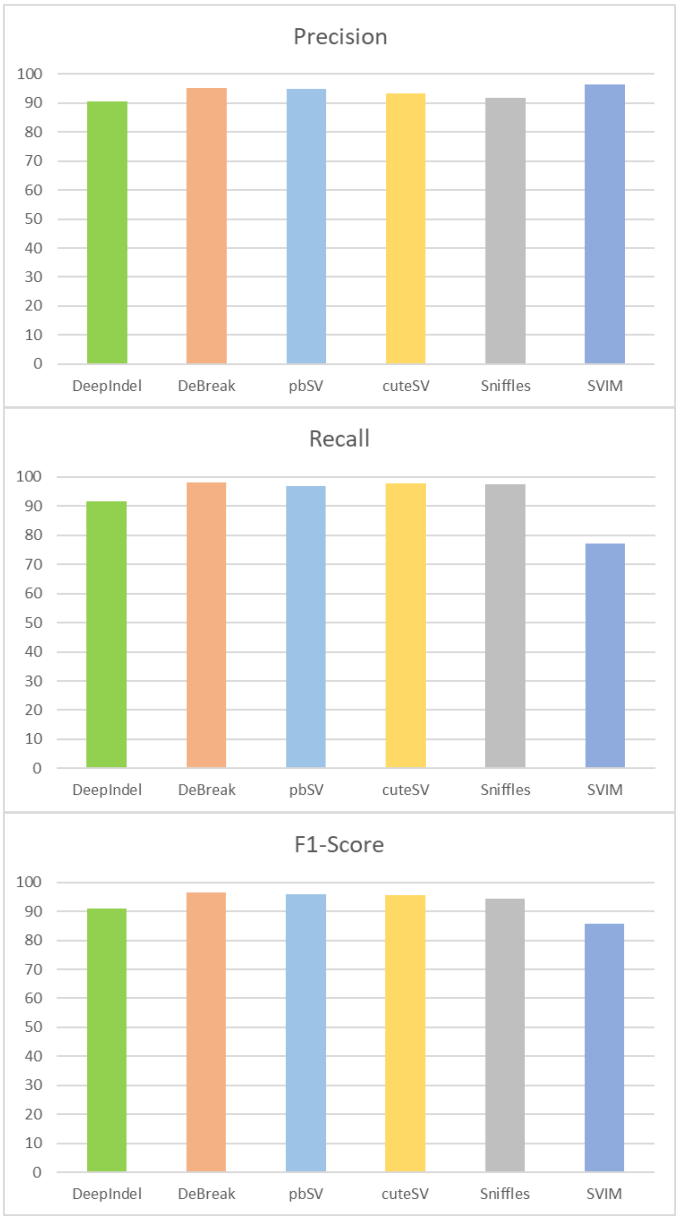


Fig. 7. Performance comparison in deletion: precision, recall, f1-score.

2) Deletion: DeepIndel exhibits strong performance in precision for deletions, achieving a precision rate of 90.42%, which is competitive with other models. For recall, DeepIndel achieves 91.76%, positioning it favorably compared to the other models. Regarding the F1-score, DeepIndel ranks well with a value of 91.09%, highlighting a good overall balance between precision and recall compared to the other models. Fig. 7 illustrates these comparisons on deletions.

V. Discussion

DeepIndel achieved outstanding accuracy in detecting SVs for the HG002 dataset, demonstrating the effectiveness and robustness of our approach. The high precision and recall rates, along with competitive F1 scores for both insertions and deletions, underscore the reliability of DeepIndel in variant calling tasks. The significance of these validation measures lies in their direct relevance to clinical utility: high precision reduces the risk of erroneous variant calls that could mislead therapeutic decisions, while robust recall ensures no critical SVs are missed in disease studies. The F1-score, achieving 94.27% for insertions and 91.09% for deletions, reflects a balanced performance that is essential for trustworthy genomic analysis. Thus, it's highlighting its strong performance overall. However, the model's performance was slightly better for insertions than for deletions. This variation in performance can be attributed, in part, to the characteristics of the benchmark dataset. The model was evaluated on the latest, most conserved BED file for HG002 (*HG002 SVs Tier1 v0.6.2.bed*), which contains 5260 insertions and 4138 deletions [22]. The higher representation of insertions in the dataset provides a richer training and evaluation context for the model, potentially contributing to its superior precision and recall for insertions. In contrast, the slightly lower performance for deletions may reflect the relatively smaller number of examples available, as well as inherent challenges in accurately identifying deletions within the genomic sequence.

While DeepIndel's performance is commendable, there is room for improvement in refining feature extraction and window positioning near breakpoint candidates to address edge cases for deletions. Nevertheless, the consistent and competitive results across various performance metrics establish DeepIndel as a highly effective tool for SV detection. The findings from the HG002 dataset confirm the robustness of DeepIndel's methodologies and its potential for broader genomic applications. Addressing minor limitations in feature extraction and ensuring robust truth datasets will further enhance the reliability and accuracy of the model.

VI. Conclusion

In this study, a novel deep learning framework, DeepIndel, was designed to enhance the accurate detection of indels from long-read sequencing data. By leveraging a comprehensive set of 23 genomic features extracted from aligned reads, the model effectively captures complex breakpoint patterns in challenging genomic regions. Trained on the HG002 benchmark dataset using a modified ResNet-50 architecture with residual connections, dropout regularization, and cross-entropy loss optimization, DeepIndel processes significant feature matrices to classify genomic sites as insertions, deletions, or non-indels with high fidelity.

DeepIndel's advancements highlight the transformative potential of integrating long-read sequencing with deep learning for SV analysis, addressing limitations in traditional methods, such as difficulties in repetitive regions and interconnected loci. By improving precision and recall, this approach facilitates more reliable insights into genomic diversity, gene expression alterations, and disease associations, including cancer, neurological disorders, and metabolic conditions. This contributes to the broader goals of precision medicine, enabling better characterization of phenotypic diversity and personalized therapeutic strategies.

Despite these strengths, some limitations and future opportunities are acknowledged that include potential refinements needed in feature extraction and window positioning to handle edge cases. Future directions will focus on expanding evaluations to diverse datasets beyond HG002, incorporating additional SV types like inversions and duplications, and optimizing for real-time clinical applications. Overall, DeepIndel marks a significant advancement in SV-calling tools, paving the way for enhanced genomic research and its translation into impactful clinical outcomes.

Code Availability

The full source code is available at this link.

Acknowledgment

We thank Dr. Atif Hasan Rahman (Associate Professor, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh) for his guidance and Dr. Kishwar Shafin (Research Scientist at Google, Health AI) for providing useful comments on our research project.

References

[1] T. G. P. Consortium, "A global reference for human genetic variation," Nature, vol. 526, pp. 68–74, 2015.

[2] R. L. Collins et al., "A structural variation reference for medical and population genetics," Nature, vol. 581, pp. 444–451, 2020.

[3] J. R. Belyeu, M. Chowdhury, J. Brown, B. S. Pedersen, M. J. Cormier, A. R. Quinlan, and R. M. Layer, "Samplot: a platform for structural variant visual validation and automated filtering," Genome Biology, vol. 22, no. 1, pp. 1–9, 2021.

[4] P. A. Audano, A. Sulovari, T. A. Graves-Lindsay, S. Cantsilieris, M. Sorensen, A. M. Welch, M. L. Dougherty, B. J. Nelson, P. Shah, S. K. Dutcher, and E. E. Eichler, "Characterizing the major structural variant alleles of the human genome," Cell, vol. 176, no. 3, pp. 663–675, 2019.

[5] P. H. Sudmant, T. Rausch, E. J. Gardner, R. E. Handsaker, A. Abyzov, J. Huddleston, Y. Zhang, K. Ye, G. Jun, M. H.-Y. Fritz, et al., "An integrated map of structural variation in 2,504 human genomes," Nature, vol. 526, no. 7571, pp. 75–81, 2015.

[6] Schloissnig, S., Pani, S., Ebler, J. et al. Structural variation in 1,019 diverse humans based on long-read sequencing. Nature 644, 442–452 (2025).

[7] S. Nurk, S. Koren, A. Rhie, M. Rautiainen, A. V. Bzikadze, A. Mikheenko, M. R. Vollger, N. Altemose, L. Uralsky, A. Gershman, S. Aganezov, S. J. Hoyt, M. Diekhans, G. A. Logsdon, M. Alonge, S. E. Antonarakis, M. Borchers, G. G. Bouffard, S. Y. Brooks, G. V. Caldas, et al., "The complete sequence of a human genome," Science, vol. 376, no. 6588, pp. 44–53, 2022.

[8] E. Espinosa, R. Bautista, R. Larrosa, and O. Plata, "Advancements in long-read genome sequencing technologies and algorithms," Genomics, vol. 116, no. 3, p. 110842, 2024.

[9] H. Li and R. Durbin, "Fast and accurate short read alignment with burrows–wheeler transform," bioinformatics, vol. 25, no. 14, pp. 1754–1760, 2009.

[10] M. Mahmoud, N. Gobet, D. I. Cruz-D´avalos, N. Mounier, C. Dessimoz, and F. J. Sedlazeck, "Structural variant calling: the long and the short of it," Genome Biology, vol. 20, p. 246, 2019.

[11] D. R. Zerbino and E. Birney, "Velvet: algorithms for de novo short read assembly using de bruijn graphs," Genome research, vol. 18, no. 5, pp. 821–829, 2008.

[12] K. Cleal and D. M. Baird, "Dysgu: efficient structural variant calling using short or long reads," Nucleic Acids Research, vol. 50, no. 9, pp. e53–e53, 2022.

[13] K. Shafin, T. Pesout, P.-C. Chang, M. Nattestad, A. Kolesnikov, S. Goel, G. Baid, M. Kolmogorov, J. M. Eizenga, K. H. Miga, et al., "Haplotype-aware variant calling with pepper-margin-deepvariant enables high accuracy in nanopore long-reads," Nature methods, vol. 18, no. 11, pp. 1322–1332, 2021.

[14] A. M. Wenger, P. Peluso, W. J. Rowell, P.-C. Chang, R. J. Hall, G. T. Concepcion, J. Ebler, A. Fungtammasan, A. Kolesnikov, N. D. Olson, et al., "Accurate circular consensus long-read sequencing improves vari- ant detection and assembly of a human genome," Nature Biotechnology, vol. 37, pp. 1155–1162, 2019.

[15] M. Jain, S. Koren, K. H. Miga, et al., "Nanopore sequencing and assembly of a human genome with ultra-long reads," Nature Biotechnology, vol. 36, pp. 338–345, 2018.

[16] F. J. Sedlazeck, P. Rescheneder, M. Smolka, H. Fang, M. Nattestad, A. von Haeseler, and M. C. Schatz, "Accurate detection of complex structural variations using single-molecule sequencing," Nature Methods, vol. 15, no. 6, pp. 461–468, 2018.

[17] R. Jian, X. Chen, and et al., "cutesv: A sensitive and fast tool for detecting short and long structural variations," Bioinformatics, vol. 36, no. 4, pp. 1233–1235, 2020.

[18] D. Heller et al., "Svim: Structural variant identification using mapped long reads," Bioinformatics, vol. 35, no. 18, pp. 2907–2915, 2019.

[19] Z. Chong et al., "Debreak: Efficiently detecting structural variations in long-read sequencing data," Genome Biology, vol. 24, no. 1, p. 14, 2023.

[20] W. Rowell et al., "Comprehensive variant detection in a human genome with highly accurate long reads," 2019. PacBio White Paper.

[21] M. Kolmogorov et al., "Nanovar: Accurate structural variant detection with ont and pacbio data," BioRxiv, 2020.

[22] N. Dierckxsens, T. Li, J. R. Vermeesch, and Z. Xie, "A benchmark of structural variation detection by long reads through a realistic simulated model," Genome Biology, vol. 22, no. 1, 2021.

[23] X. Zhang, N. Han, and J. Zhang, "Comparative analysis of vgg, resnet, and googlenet architectures evaluating performance, computational efficiency, and convergence rates," Applied and Computational Engineering, vol. 44, pp. 172–181, 03 2024.

[24] S. Mascarenhas and M. Agarwal, "A comparison between vgg16, vgg19 and resnet50 architecture frameworks for image classification," in 2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON), vol. 1, pp. 96–99, 2021.

[25] N. Zakaria, F. Mohamed, R. Abdelghani, and K. Sundaraj, "Vgg16, resnet-50, and googlenet deep learning architecture for breathing sound classification: A comparative study," in 2021 International Conference on Artificial Intelligence for Cyber Security Systems and Privacy (AI- CSP), pp. 1–6, 2021.

[26] H. Li, "Minimap2: pairwise alignment for nucleotide sequences," Bioinformatics, vol. 34, no. 18, pp. 3094–3100, 2018.

[27] H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, R. Durbin, and . G. P. D. P. Subgroup, "The sequence alignment/map format and samtools," bioinformatics, vol. 25, no. 16, pp. 2078–2079, 2009.

[28] N. Dierckxsens, T. Li, J. R. Vermeesch, et al., "A benchmark of structural variation detection by long reads through a realistic simulated model," Genome Biology, vol. 22, no. 342, 2021.

[29] H. Elrick, C. M. Sauer, J. Espejo Valle-Inclan, K. Trevers, M. Tanguy, S. Zumalave, S. De Noon, F. Muyas, R. Cascão, A. Afonso, A. G. Rust, F. Amary, R. Tirabosco, A. Giess, T. Freeman, A. Sosinsky, K. Piculell, D. T. Miller, C. C. Faria, G. Elgar, *et al.*, "SAVANA: reliable analysis of somatic structural variants and copy number aberrations using long-read sequencing," *Nature Methods*, vol. 22, no. 7, pp. 1436–1446, 2025.

[30] R. Li, H. Chu, K. Gao, H. Luo, and Y. Jiang, "SUMMER: an integrated nanopore sequencing pipeline for variants detection and clinical annotation on the human genome," *Functional & Integrative Genomics*, vol. 25, no. 1, p. 21, 2025.

[31] T. Jiang, Z. Zhou, Z. Zhang, S. Cao, Y. Wang, and Y. Liu, "MEHunter: transformer-based mobile element variant detection from long reads," *Bioinformatics*, vol. 40, no. 9, p. btae557, 2024.

[32] A. Daigle, L. S. Whitehouse, R. Zhao, J. J. Emerson, and D. R. Schrider, "Leveraging long-read assemblies and machine learning to enhance short-read transposable element detection and genotyping," *bioRxiv*, p. 2025.02.11.637720, 2025.

[33] H. Ding and J. Luo, "MAMnet: detecting and genotyping deletions and insertions based on long reads and a deep learning approach," *Briefings in Bioinformatics*, vol. 23, no. 5, p. bbac195, 2022.

[34] Y. Zheng and X. Shang, "SVcnn: an accurate deep learning-based method for detecting structural variation based on long-read data," *BMC Bioinformatics*, vol. 24, no. 1, p. 213, 2023.

A Modeling Approach for Strategic Fleet Sizing Under Maritime Sovereignty: Application to the Moroccan National Fleet

Mohamed Anas KHALFI[1], Aziz AIT BASSOU[2], Mustapha HLYAL[3], Jamila EL ALAMI[4]
LASTIMI Laboratory, High School of Technology in Sale, Mohammed V University in Rabat, Rabat, Morocco[1, 2, 4]
Center of Excellence in Logistics, Higher School of Textile and Clothing Industries, Casablanca, Morocco[3]

***Abstract*—The disruptions experienced by global supply chains in recent years have reignited the importance of maritime sovereignty, particularly through the creation or reinforcement of national shipping fleets. In this context, the present study explores strategic approaches to national fleet sizing, drawing from recent policy directions and maritime planning models. The study is motivated by the need to design resilient and sovereign fleets that reduce dependency on foreign operators and strengthen autonomy in trade logistics. To complement this analysis, a mathematical model is developed in the form of a Mixed-Integer Nonlinear Programming (MINLP) formulation, where sovereignty is captured through the share of vessel operations under national control. In addition to sovereignty, the model integrates criteria of economic viability, environmental impact, and resilience, positioning Maritime Fleet Sizing within the broader scope of Strategic Transport Planning and Green Maritime Transport. Numerical experiments are carried out on a representative dataset of vessels and strategic routes, illustrating how sovereignty thresholds affect fleet composition and deployment. The results highlight a fundamental trade-off between sovereignty and profitability, emphasizing the need for strategic decision-making that carefully balances autonomy objectives with resilience and environmental considerations. Findings also show that moderate sovereignty thresholds support cost-efficient and diversified fleets, while maximalist sovereignty requirements lead to reduced coverage, higher unmet demand, and lower profitability. These insights underline the importance of calibrated strategies, where Sovereignty, Resilience in Maritime Logistics, and sustainability are treated as interconnected pillars of long-term fleet development.**

Keywords*—*Maritime fleet sizing; sovereignty and national ownership; strategic transport planning; resilience in maritime logistics; green maritime transport

I. INTRODUCTION

Africa's maritime domain represents a major strategic asset, essential for economic development and regional integration [1]. Nevertheless, this potential remains largely untapped due to structural vulnerabilities: inadequate port infrastructure, dependence on foreign shipping lines, weak logistical connectivity, and growing security threats. In this context, establishing sovereign national fleets appears as a critical lever to secure economic autonomy and strengthen the trade resilience of African states. Yet, recent studies reveal that the intersection between sustainability, resilience, and sovereignty in supply chain planning remains insufficiently explored, particularly when considering the role of local product promotion and national industrial strategies [2].

The experiences of Nigeria and Ghana during the 1960s to 1980s illustrate the difficulties associated with establishing national shipping companies. These initiatives aimed to reduce dependency on European maritime powers but largely failed due to poor governance, systemic corruption, and the inability to modernize fleets in the face of containerization [3] [4]. The primary lesson for developing countries is the need to avoid politicization in maritime management and to sustain continuous technological upgrading.

The major lesson to be learned by developing countries is that they should avoid politics in the management of the maritime industry and that they should ensure a consistent technological updating. Models of governance that have been the most effective in key worldwide ports have been those that are publicly-owned but privately managed (landlord model) [5] [6]. Excessive privatization makes the ports vulnerable to strategic losses, whereas inflexible governance restrains innovation. The model applicable in developing countries should be a hybrid model that favors the public-private partnerships and still maintains its sovereignty of strategic assets. In this context, Build-Operate-Transfer (BOT) schemes also represent a viable pathway, allowing private actors to finance and operate vessels under sovereign constraints before eventual transfer to public ownership.

The competitiveness of a national fleet is closely linked to the performance of port infrastructures. In [7], the authors demonstrate that improving connectivity and port quality significantly reduces freight costs, facilitating integration into global supply chains. For Morocco, strengthening Tanger Med, Nador West Med, and Casablanca is crucial to support the future sovereign fleet.

An integrated maritime security system, combining surveillance, environmental governance, and regional cooperation, is critical to sustain Morocco's ambitions in the blue economy, and it can only be achieved through the control of economic flows and the sovereignty of its maritime trade [8] [9].

The comparative analysis of African and international experiences shows that the success of a national maritime fleet depends on smart hybrid governance, high-performance infrastructure, integration into global supply chains, and

comprehensive maritime security. Morocco's dual maritime façade (Atlantic-Mediterranean interface), its port assets (Tanger Med), and its proximity to the Strait of Gibraltar make it a natural maritime nation, assuming responsibilities as a coastal, port, and flag state [10]. Yet, strategic control over shipping remains fragile in the absence of a national fleet capable of withstanding foreign dependency. As of 2019, nearly 98 per cent of Morocco's external trade depended on maritime transport, while the potential collapse of national operators like IMTC was projected to result in complete reliance on foreign-owned fleets [11].

The key objective of this study is to make a contribution to the strategic planning of the maritime fleet through a structured modeling approach that puts the sovereignty parameter at its center. Indeed, this study will seek to offer a mathematical model applicable to the strategic development of a national commercial fleet, mainly adapted to developing countries such as Morocco.

This study is structured as follows: Section II draws a literature review that covers the approaches that have been taken in relation to the maritime fleet size and mix, typologies of methods used, and gaps that can be addressed through the strategies that are policy-oriented. Section III introduces the proposed optimization model, detailing its decision variables, parameters, and strategic criteria, including cost, sovereignty, environmental impact, and resilience. Section IV discusses the simulation framework, while Section V presents the model's results, and offers analysis and adequate interpretation, mainly regarding the sovereignty parameter, profitability, sustainability and resilience constraints. Section VI concludes and outlines future work.

II. Literature Review

The maritime fleet size and mix problem (MFSMP) constitutes a core research theme in operations research and maritime economics, addressing how to determine the optimal number and types of vessels needed to meet transportation demand under a variety of economic, operational, and regulatory constraints [12]. In the context of emerging economies such as Morocco, where maritime sovereignty, trade resilience, and infrastructure modernization are national priorities, this body of research offers valuable, yet incomplete, tools to guide long-term strategic planning.

Initial work in the area was based on deterministic and static formulations, such as [13]and [14], with optimization of fleet composition usually through linear programming approaches based on the assumption that demand is fixed and that ship types are uniform. Although such models were analytically tractable, they did not deal with long-term investment options, changing market circumstances or maritime lifecycle behavior.

A conceptual shift occurred with the development of multi-period models, commonly described as maritime fleet renewal problems (MFRP). These models are thoroughly discussed in [12], which differentiates between single-period static designs and long-horizon fleet strategies that consider aging of the ships, investment cycles, and changing operational demands. Their work also signals the significance of adaptability to fleet planning, in states wanting to undertake fleet renewal within the context of more general industrial and economic development. In relation to the case of Morocco, the national fleet has fallen from about 70 vessels in the 1980s to less than 20 in the 2020s, which has been seen as a problem of growing size in terms of structural and persistent decline [11]. New developments have focused on the inclusion of uncertainty in models used to size the fleet. Both [15] and [16] have suggested recourse to stochastic programming with regard to variability in charter rates, bunker costs, and demand. These models show the superiority of flexibility in investment and chartering decisions with volatile conditions. In addition to this, [17] presented a rule-based simulation framework to deal with uncertainty in the process of buying and selling and formed a solid foundation of practical application of adaptive maritime policy-making.

The literature has also seen the rise of heuristic and metaheuristic methods, such as genetic algorithms, particle swarm optimization [18], and tabu search, to handle complex routing and heterogeneous fleet structures. While effective in solving tactical-level problems, these approaches often lack a strategic orientation necessary for national-level fleet planning.

From a governance perspective, [19] proposed a typology distinguishing strategic, tactical, and operational fleet sizing decisions. Their classification is of some significance to states such as Morocco, which have to balance long-term capital intensity, infrastructural coordination and policy-driven maritime goals in their fleet-based decisions. Their work also describes structural differences in transportation planning between maritime and land transport planning. Despite these contributions, there are still a number of limitations. The majority of such existing models are oriented towards optimality in the case of firms and are unsuitable to reflect the particular needs of the public sector, which deals with maritime sovereignty, autonomy in trade or industrial capacity building. Also, lack of preset standards constrains cross-case comparisons and dampens the process of transferring modeling knowledge across borders or nationwide.

Table I, Table II and Table III support the fact that there is a need to contribute more to the modelling literature that puts the national policy applicability of results as a main criterion. In Morocco, whose port facilities are developing rapidly, maritime commerce is one of its main strategic platforms. Thus, fiscal investment limitations, dynamic demand, and nationwide governance of maritime logistics require an extensive adjustment of fleet size templates. Such models will be instrumental in informing the design of a resilient, efficient, and sovereign commercial fleet aligned with the country's long-term development trajectory. However, previous legislative reforms have been criticized for their lack of fiscal incentives and failure to embed the Moroccan economic interest, thereby weakening the strategic effectiveness of the national fleet [10].

TABLE I. MODELING-ORIENTED CONTRIBUTIONS

Reference	Transport Type	Modeling Method	Mathematical Approach	Time Horizon	Decision Level	Criteria Included	Geographic Scale	Relevance to Moroccan Case
[17]	Liner Shipping	Simulation + Scenario Analysis	Genetic Algorithms	Short to Mid-term	Tactical	Cost, Uncertainty	Regional (Japan)	Medium
[18]	Offshore / Arctic	Fleet Mix Optimization	Particle Swarm Optimization	Short to Mid-term	Tactical to Strategic	Cost, Uncertainty, Capacity	Arctic (Norway)	Medium
[20]	Industrial and Bulk shipping	Robust Integer Programming	MIP	Mid-term	Tactical	Cost, Capacity, Market	Global	High
[21]	Liner Shipping	System Dynamics	Qualitative + Simulation	Short- term	Tactical	Profit, Carbon emission	Global	High
[22]	Liner Shipping	Stochastic Optimization	Two-Stage SP + Scenario	Long-term	Strategic	Uncertainty, Cost	Global	High
[23]	Liner Shipping	Simulation Models	System Dynamics	Long-term	Strategic	Capacity, Sovereignty	National (Dev. Countries)	High
[16]	Liner Shipping	Scenario-based dynamic programming	MILP	Short to Mid-term	Tactical	Cost, Time	Global	High
[24]	Liner Shipping	Strategic routing problems	Linear Programming / MIP	Long-term	Strategic	Cost, Demand	Global	High
[14]	US Navy	Fleet Planning	Integer Programming	Long-term	Strategic	Cost, Port constraints	National (USA)	Low
[25]	Naval/Military	Strategic Lift Modeling/ AHP	MIP	Long-term	Strategic	Cost, Operational Constraints	National (USA)	Low
[26]	Ship Chartering	Strategic Deployment	Mathematical Optimization	Long-term	Strategic	Cost, Logistics	Global	Low
[27]	Offshore Supply	Routing and Scheduling	Integer Programming	Short-term	Tactical	Cost, Efficiency	National (Norway)	Medium
[28]	Large containerships	Costing Models	DEA	Mid-term	Tactical	Efficiency	Global	Low
[29]	Liner Shipping	Hub & Spoke scheduling	MIP	Short-term	Tactical	Cost, Time	South East Asia	Low
[30]	Liner Shipping	Routing + Fleet Sizing	Set Partitioning + Routing	Short-term	Tactical	Cost, Efficiency	Regional (Norway)	Medium
[31]	Liner Shipping	Network Design Problem	MIP + Heuristics	Long-term	Strategic	Cost, Competition	Global	Medium
[32]	Liner Shipping	Stochastic	Set Partitioning + Heuristics	Long-term	Strategic	Cost, Emission	Firm-level	Medium

TABLE II. POLICY AND STRATEGIC PLANNING CONTRIBUTIONS

Reference	Transport Type	Modeling Method	Mathematical Approach	Time Horizon	Decision Level	Criteria Included	Geographic Scale	Relevance to Moroccan Case
[1]	All types of maritime transportation	N/A	N/A	Long-term	Strategic	Sovereignty, Infrastructure	Regional (East & South Africa)	High
[33]	Coastal Shipping	Case Study + Policy Analysis	N/A	Mid to Long-term	Strategic	Sovereignty, Security	Regional (Sub-Saharan Africa)	High
[8]	All types of maritime transportation	N/A	N/A	Long-term	Strategic	Security, Sovereignty	Continental (Africa)	Medium
[34]	All types of maritime transportation	N/A	N/A	Long-term	Strategic	Sovereignty, Governance	Continental (Africa)	Medium
[4]	All types of maritime transportation	N/A	N/A	Long-term	Strategic	Sovereignty, Governance	Continental (Africa)	Medium
[35]	All types of maritime transportation	N/A	N/A	Long-term	Strategic	Policy & Infrastructure Review	National (Algeria)	Medium
[6]	All types of maritime transportation	N/A	N/A	Long-term	Strategic	Policy & Infrastructure Review	Global	Medium

[36]	All types of maritime transportation	N/A	N/A	Long-term	Strategic	Security, Resilience	Regional (Caribbean)	High
[37]	All types of maritime transportation	Discourse Analysis	N/A	Long-term	Strategic	GHG, Policy Framing	Regional (EU)	High
[38]	Naval/Defense	N/A	N/A	Long-term	Strategic	Resilience, Sovereignty	National (USA)	Medium
[23]	Liner Shipping	Simulation Models	System Dynamics	Long-term	Strategic	Capacity, Sovereignty	National (Dev. Countries)	High
[39]	All types of maritime transportation	N/A	N/A	Long-term	Strategic	Flagging, Sovereignty	National (Germany)	Low
[3]	Liner Shipping	N/A	N/A	Long-term	Strategic	Cost, Utilization	Regional (West Africa)	Medium
[7]	Liner Shipping	N/A	N/A	Long-term	Strategic	Policy & Infrastructure	Regional (Caribbean)	Low
[5]	All types of maritime transportation	N/A	N/A	Long-term	Strategic	Policy & Infrastructure Review	Global	Low
[40]	Naval/Defense	Agent-Based Modeling	Simulation	Mid to Long-term	Strategic	Behavioral, Capacity	National (Indonesia)	Low
[41]	Commercial Fleet	Empirical Performance Analysis	Panel Regression	Long-term	Strategic	Fleet Size, Profitability	National (Korea)	Medium

TABLE III. RESILIENCE, UNCERTAINTY, AND SUSTAINABILITY CONTRIBUTIONS

Reference	Transport Type	Modeling Method	Mathematical Approach	Time Horizon	Decision Level	Criteria Included	Geographic Scale	Relevance to Moroccan Case
[42]	Tramp and Industrial Shipping	Optimization and simulation	Monte Carlo Simulation	Long-term	Strategic	Cost, Uncertainty	National (Norway)	High
[17]	Liner Shipping	Simulation + Scenario Analysis	Genetic Algorithms	Short to Mid-term	Tactical	Cost, Uncertainty	Regional (Japan)	Medium
[43]	Container Shipping	N/A	N/A	N/A	N/A	Cost, Emissions	Atlantic Port cities	Medium
[18]	Offshore / Arctic	Fleet Mix Optimization	Particle Swarm Optimization	Short to Mid-term	Tactical to Strategic	Cost, Uncertainty, Capacity	Arctic (Norway)	Medium
[15]	Industrial /chemical Shipping	N/A	N/A	Mid to Long-term	Tactical to Strategic	Cost, Uncertainty	Arctic (Norway)	High
[44]	All types of maritime transportation	Futures Analysis	Foresight & System Dynamics	Long-term	Strategic	Resilience, Sustainability	Continental (Africa)	Medium
[37]	All types of maritime transportation	Discourse Analysis	N/A	Long-term	Strategic	GHG, Policy Framing	Regional (EU)	High
[45]	All types of maritime transportation	Resilience Index Analysis	N/A	Long-term	Strategic	Resilience, Emissions, security	National (China)	Low
[46]	Liner Shipping	Graph Theory	Non Linera Programming	Long-term	Strategic	Connectivity, Resilience	Global	High
[47]	Industrial /chemical Shipping	Stochastic Optimization	Two-stage Stochastic optimization	Long-term	Strategic	Ageing (retrofit), GHG emissions	Global	High
[22]	Liner Shipping	Stochastic Optimization	Two-Stage SP + Scenario	Long-term	Strategic	Uncertainty, Cost	Global	High
[38]	Naval/Defense	N/A	N/A	Long-term	Strategic	Resilience, Sovereignty	National (USA)	Medium
[40]	Naval/Defense	Agent-Based Modeling	Simulation	Mid to Long-term	Strategic	Behavioral, Capacity	National (Indonesia)	Low
[36]	All types of maritime transportation	N/A	N/A	Long-term	Strategic	Security, Resilience	Regional (Caribbean)	High
[32]	Liner Shipping	Stochastic	Set Partitioning + Heuristics	Long-term	Strategic	Cost, Emission	Firm-level	Medium

III. MODELING FORMULATION

This section presents the mathematical formulation of the model developed to support sovereign fleet planning. Built as a strategic decision-support tool, the model aims to reflect the multifaceted priorities of public authorities; balancing profitability with national interest, environmental compliance, and operational robustness. Unlike conventional formulations rooted in commercial fleet management [24][48][12], this approach is shaped by the realities of maritime policy in emerging economies, as the renewal of their national fleets requires a long-term framework that integrates logistical needs, industrial ambitions, and environmental constraints [10]. The model developed in this study is specifically tailored to the context of liner shipping operations, where vessels operate on fixed schedules and recurring routes. Other maritime segments, such as tramp shipping or industrial bulk transport, are outside the scope of this research. Indeed, the model simultaneously:

- maximizes the sovereign-adjusted profit generated by the fleet;
- penalizes unmet transport demand to enhance system resilience;
- and internalizes carbon taxation within the economic decision process.

Such an integrated approach responds to critical gaps identified in the literature: the lack of sovereign decision-making frameworks [8] [10] [35], the fragmented treatment of resilience [20][48][18], and the peripheral role of environmental concerns in traditional fleet models [16] [49] [37].

Thus, the model was built in layers, following the logic of national maritime planning. The priority was to simulate investment decisions under sovereign control, that is, to determine which types of vessels should be included in the national fleet, and in what configurations. This led to the formulation of the activation variable y_i, allowing each vessel i to be activated in the fleet, and when combined with the parameter A_{ik}, it reflects the possible configurations like feeder, regional, or deep-sea cargo capacity.

Once vessels were activated, the next layer addressed route assignment. Here, the binary variable x_{ir} determines whether vessel i is assigned to route r, reflecting the planning logic of matching vessel profiles to strategic corridors.

Finally, the third decision layer determines transport capacity deployed, with continuous variable q_{ir} representing the number of containers shipped by each vessel on each route. This variable is bounded by the technical capacity of the vessel, and its aggregation at the route level is compared against demand D_r.

Each of these layers is supported by constraints ensuring investment realism, operational feasibility, and policy compliance (e.g., emission limits). The result is a three-tiered model as illustrated in Fig. 1:

- Activation → strategic planning
- Assignment → operational deployment
- Transport → economic performance

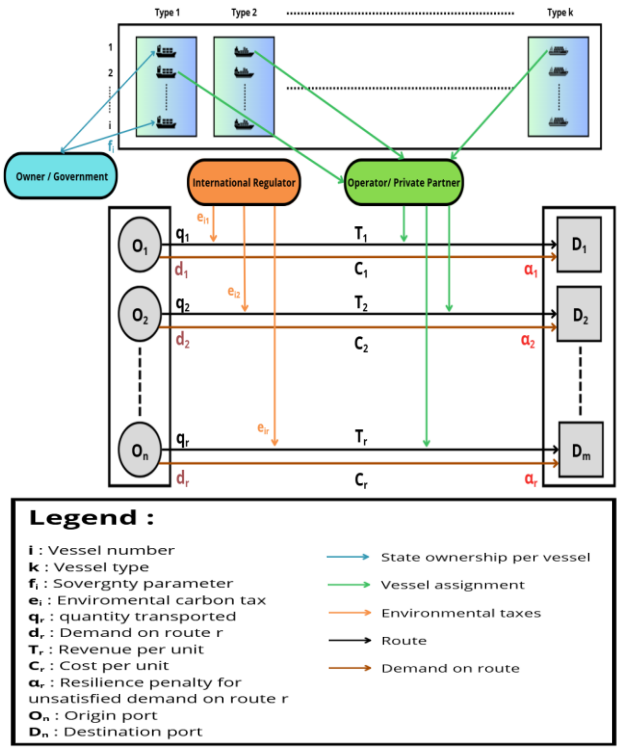


Fig. 1. The conceptual model of strategic fleet sovereignty model.

A. Sets and Variables

Let the following sets and parameters be defined:

Sets:

- I: Set of candidate vessels
- K: Set of vessel types (e.g., feeder, post-panamax)
- R: Set of strategic maritime routes

Parameters:

- T_{ir}: Revenue per TEU (Twenty-foot Equivalent Unit) for vessel i on route r
- C_{ir}: Operating cost per TEU for vessel i on route r
- e_{ir}: Carbon cost or penalty per TEU for vessel i on route r
- Q_k: Maximum transport capacity of vessel type k
- f_i : Sovereign ownership or financing ratio of vessel i
- θ : The minimum required average sovereignty ratio for the activated fleet
- α_r : Penalty for unsatisfied demand on route r
- Φ_i: Fixed operating cost for vessel i, incurred only if the vessel is activated.
- d_r : Transport demand on route r (TEUs)
- I_{ik}: Investment cost to activate vessel i as type k

- A_{ik}: A binary parameter equal to 1 if vessel i is from type k
- B : Total investment budget
- E^{max} : Maximum allowable CO_2 emissions

The model's decision variables include:

- $y_i \in \{0,1\}$: 1 if vessel i is activated
- $x_{ir} \in \{0,1\}$: 1 if vessel i is assigned to route r
- q_{ir}: Volume (in TEUs) transported by vessel i on route r

B. Justification of the Sovereignty Coefficient f_i by Vessel type

In order to reflect the importance of maritime sovereignty in the decision-making process of the fleet size and mix problem, the model presents a sovereignty coefficient to each vessel. This coefficient ranges from 0 to 1 and reflects the extent of governmental ownership for each vessel. Higher values indicate a stronger alignment with national sovereignty objectives. These values were determined based on typical ownership patterns, operational zones, and investment accessibility. Table IV summarizes the estimated sovereignty coefficients by vessel type.

TABLE IV. THE ESTIMATED SOVEREIGNTY COEFFICIENTS BY VESSEL TYPE

Vessel Type	f_i	Justification
Feeder	0.90	Frequently used in short-distance regional shipping; often owned or operated by local or national entities, enabling high sovereign control.
Small Handy	0.88	Small-scale vessels suitable for secondary routes, they are commonly accessible to domestic operators or public investment initiatives.
FeederMax	0.85	Larger than standard feeders, they are often deployed in extended regional services, typically still under partial or full national control
Panamax	0.70	Moderate capacity vessels able to transit the Panama Canal. Some national fleets operate them, though many are controlled by international shipping lines.
Medium Containership	0.65	Typical in mid-range logistics, they are widely deployed by global carriers but still present in some sovereign or mixed-ownership fleets
Post-Panamax	0.60	Larger vessels mostly integrated into the global fleet, they are generally inaccessible to smaller national actors.
Neo-Panamax	0.55	Designed for the expanded Panama Canal, their ownership and deployment are dominated by large international companies.
ULCV (Ultra Large Container Vessel)	0.40	Among the largest vessels globally (>20,000 TEUs), they are typically owned by a few multinational giants, making sovereign control highly unlikely

C. The Objective Function

This objective function captures the sovereignty-weighted profit, penalizes unmet demand to simulate resilience, and internalizes the investment cost of sovereign fleet expansion. The introduction of f_i as a decision-dependent parameter transforms the model from a cost-efficiency tool into a policy-aligned mechanism for state-led fleet control. This approach directly addresses gaps noted by [8] [21][44], who highlight the lack of sovereign metrics in operational maritime models. The objective function is defined as:

$$Max \sum_{i \in I} \sum_{r \in R} f_i \,.\, x_{ir} .\, (q_{ir} \,.\, (T_{ir} - C_{ir}) - \sum_{i \in I} \Phi_i \cdot y_i - \sum_{i \in I} \sum_{r \in R} e_{ir} .\, x_{ir} - \sum_{r \in R} \alpha_r \,.\, (d_r - \sum_{i \in I} q_{ir}) - \sum_{i \in I} \sum_{k \in K} f_i \,.\, \mathrm{I}_{ik} \,.\, A_{ik} .\, y_i$$

The objective function combines five components that consider the strategic, economic and environmental priorities of the model. The first term maximizes revenue on the transported volumes, weighted by the ratio of the sovereign ownership, hence encouraging national control over the maritime activities. This form of structural representation of sovereignty covers one of the key deficiencies of the traditional form of fleet sizing models under the assumption of neutrality in ownership structure [14][4][1]. The second term reflects the fixed costs of operation that are incurred when a vessel is activated, which is a consideration to make the fleet composition selective. The third term is about the carbon charges imposed on routes so as to discourage environmental externalities and make sure that environmental concerns are fulfilled based on deployment considerations. The fourth term will impose the demand unmet as a penalty, which aligns with recent findings highlighting how managing demand uncertainty through cooperative and sustainable sourcing strategies can strengthen resilience, especially in contexts aiming for national or regional autonomy [50]. Lastly, the fifth term, which is the sovereign investment costs, scaled by the ownership ratio, reflects the financial implications of an increase in capacity controlled by the government.

The objective function includes a bilinear term of the form:

$$\sum_{i \in I} \sum_{r \in R} f_i \,.\, x_{ir} .\, (q_{ir} \,.\, (T_{ir} - C_{ir})$$

This term contains the product of the binary decision variable x_{ir} and the continuous variable q_{ir}. This nonlinearity cannot be directly handled by linear optimization solvers. To address this, an auxiliary variable Z_{ir} is introduced to represent the product:

$$Z_{ir} = x_{ir} .\, q_{ir}$$

This variable is then linked to x_{ir} and q_{ir} through the following linear constraints, involving a sufficiently large constant M (Representing the upper bound of transported volume q_{ir}):

$$\begin{cases} Z_{ir} \le q_{ir}, \forall\, i \in I, r \in R \\ q_{ir} \le M .\, x_{ir}, \forall\, i \in I, r \in R \\ Z_{ir} \le M .\, x_{ir}, \forall\, i \in I, r \in R \\ Z_{ir} \ge q_{ir} - M(1 - x_{ir}), \forall\, i \in I, r \in R \end{cases}$$

The original bilinear expression in the objective function is then replaced by the linear term:

$$\sum_{i \in I} \sum_{r \in R} f_i \,.\, Z_{ir} \,.\, (T_{ir} - C_{ir})$$

D. Constraints

The model is subject to the following strategic and operational constraints:

1) Single Route Assignment per Vessel

$$\sum_{r \in R} x_{ir} \leq 1, \forall i \in I$$

This constraint ensures that a vessel is not assigned to more than one route, simultaneously.

2) Vessel activation condition

$$x_{ir} \leq y_i \;\; \forall\, i \;\in\; I, r \;\in\; R$$

Prevents deployment of vessels not activated [51].

3) Capacity Limit by Type

$$q_{ir} \leq \sum_{k \in K} Q_k . A_{ik} . y_i \;\; \forall\, i \;\in\; I, r \;\in\; R$$

Restricts capacity usage to actual vessel capability, as in [16] and [39].

4) Partial Demand Coverage (Resilience)

$$\sum_{i \in I} q_{ir} \leq d_r \;\; \forall\, r \;\in\; R$$

Allows under-supply, simulating strategic bottlenecks. This model's resilience is formalized by [20] and [48].

5) Investment Budget

$$\sum_{i \in I} \sum_{k \in K} f_i . \mathrm{I}_{ik} . A_{ik} . y_i \leq B \;\; \forall\, r \;\in\; R$$

Controls sovereign expenditure [22].

6) Environmental Emission Cap

$$\sum_{\boldsymbol{i \in I}} \sum_{\boldsymbol{r \in R}} \boldsymbol{e_{ir} . x_{ir} \leq E^{max}}$$

Enables regulatory compliance and green planning strategies [37] and [36].

7) Sovereignty

$$\sum_{i \in I} f_i \, y_i \geq \theta \sum_{i \in I} y_i$$

8) Variable Domains

$$y_i \in \{0,1\}, x_{ik} \in \{0,1\}, q_{ir} \geq 0$$

E. Assumptions

The proposed model relies on a series of simplifying assumptions aimed at ensuring both analytical tractability and strategic relevance. These assumptions are consistent with the intended use of the model as a strategic planning tool for long-term maritime policy design. First, the model is developed as a deterministic, single-period formulation, representing a strategic decision snapshot over a long-term planning horizon (10 years or more). This choice reflects the irreversible and capital-intensive nature of fleet investment decisions, where vessel acquisition and sovereign activation are made on a long-term basis and are not subject to frequent revision. Similar simplifications are made in strategic logistics models dealing with infrastructure design or energy fleet mix [12] [44]. Second, transport demand on each route $d_r$is treated as known and deterministic. This is justified by the model's focus on national routes of strategic importance (e.g., connecting major ports or corridors), for which long-term projections are available from port authorities or national development plans. This assumption avoids the complexity of scenario modeling or stochastic programming, while still allowing partial satisfaction of demand via the resilience penalty. Third, the environmental taxation is modeled through a fixed carbon cost per TEU e_{ir} , which is not influenced by minor variations in vessel load. This simplification acknowledges that while actual emissions vary with loading conditions and speed profiles [49], policy-applied taxes and penalties often use standardized rates per unit transported, as recommended in IMO and EU carbon schemes. This assumption is reasonable for strategic planning where environmental constraints are treated as policy ceilings rather than engineering simulations [37] [36]. Then, we assume that each vessel can be assigned to at most one route, and its transport volume q_{ir} is dedicated entirely to that route. This assumption reflects the model's focus on permanent or seasonal deployment planning, rather than flexible daily scheduling. It is particularly relevant for state-led services (e.g., national supply corridors, food security logistics) where vessels are dedicated based on geopolitical or regional priorities [8] [18]. Also, the model does not assume any preferential match between vessel types and demand profiles. Instead, vessel type selection (via y_{ik}) and route assignment (via x_{ir}) are endogenously determined by the optimization process. This allows for flexibility in configuring the fleet to meet evolving transport and policy requirements. Finally, all costs (investment, operational, and environmental) and revenues are either modeled linearly or linearized using linearization techniques. This allows the use of mixed-integer linear programming (MILP) techniques, which are well-established in fleet sizing literature and offer reasonable solvability even for large-scale problems [48] [12].

IV. Simulation Framework

The simulations were carried out on a typical desktop personal computer with Intel core i7 1.9 GHz and 16 GB of RAM. Pyomo modeling framework was used to write the optimization model in Python, and used SCIP 9.2.2, an open-source high performance solver for mixed-integer linear programs developed by the Zuse Institute Berlin (ZIB). It is a compilation of the solver added in optimized mode, 8-byte precision and based on SoPlex 7.1.4 as its LP backend.

Importantly, the Branch & Cut algorithm was employed to ensure the derivation of exact optimal solutions, reinforcing the model's credibility for strategic-level decision-making.

The computational tests were based on a realistic configuration involving 100 candidate vessels, 230 strategic maritime routes, and 8 vessel types, as per Table IV. The choice of 100 vessels and 230 routes is justified by the ambition of Morocco, as an example of an emerging country, to rebuild its national maritime fleet by 2040. The full model formulation resulted in a total of 69,100 variables, of which 23,100 are binary and 46,000 continuous, underlining the combinatorial complexity of the problem. To evaluate the model's behaviour under different sovereignty priorities, four simulations were performed by varying the minimum average sovereignty threshold parameter θ:

- Low requirement ($\theta = 0.25$)
- Moderate requirement ($\theta = 0.50$)
- High requirement ($\theta = 0.75$)
- Sovereignty-maximalist ($\theta = 0.88$)

During every run, the model produced the most favorable sizes of the fleet, the composition of the vessels, route assignments, and volumes transported. Randomly generated values of inputs and in calibrated bounds based on real maritime benchmarks have been considered without violation of investment, capacity, emission and policy constraints. An observation of the computational performance was made in every scenario. The calculation time was between: 330.61 seconds ($\theta = 0.25$), 401.49 seconds ($\theta = 0.50$), 402.91 seconds ($\theta = 0.75$) and 549.99 seconds ($\theta = 0.88$), denoting increases in time needed to solve based on severe sovereignty constraints.

V. Results

In order to evaluate the model's response to real-life problems, different scenarios were examined to highlight the trade-offs between cost and fleet sovereignty, environmental influences and resilience. The analysis reveals the way in which changes of strategic priorities alter the optimal structure and utilization of the fleet. It also sheds light on how the transport demand and environmental aspects are impacted.

A. Fleet Mix, Size and Activation Patterns

The optimization outcomes indicate a similar pattern in the vessel activation decisions in all scenarios, as shown in Table V. At best, the model prefers a smaller, more effective fleet mix that is mainly composed of mid-sized vessels: Feeder, Medium CS, and Small Handy under a smaller threshold of sovereignty ($\theta = 0.25$ and $\theta = 0.50$). As θ increases, reflecting higher sovereignty constraints, the diversity of activated vessels drops drastically, and total activation volume declines, particularly on high-volume routes.

When $\theta = 0.88$, the model heavily prioritizes vessels with the highest sovereignty scores, even at the cost of reduced transport capacity and higher unmet demand. Notably, the vessels of type ULCV and Panamax family (Panamax, Post-Panamax and Neo-Panamax) are consistently excluded across all scenarios.

Regarding the fleet size, Table VI illustrates the evolution of total fleet size per scenario. When $\theta = 0.25$ or $\theta = 0.50$, the model result is respectively a fleet of 17 vessels that are cost-efficient and sovereignty sensitive. The number of activated vessels increases significantly to 53 when $\theta = 0.75$, suggesting a diversification strategy that covers a wider range of demand. However, the number of activated vessels decreases to 18 once more when the sovereignty constraint reaches $\theta = 0.88$.

TABLE V. Activated Vessel Types per Sovereignty Scenario

Vessel Type		Sovereignty Score (f_i)	θ = 0.25	θ = 0.50	θ = 0.75	θ = 0.88
Feeder		0.90	Yes	Yes	Yes	No
Feedermax		0.85	No	No	No	No
Small Handy		0.88	Yes	Yes	Yes	Yes
edium CS		0.65	Yes	Yes	Yes	No
Panamax		0.70	No	No	No	No
Post-Panamax		0.60	No	No	No	No
Neo-Panamax		0.55	No	No	No	No
ULCV		0.40	No	No	No	No

TABLE VI. Total Number of Activated Vessels Per Scenario

Sovereignty Threshold (θ)	Number of Activated Vessels	Comments
0.25	17	The fleet remains compact
0.50	17	The fleet is almost as per the scenario θ = 0.25
0.75	53	High increase in fleet size
0.88	18	Many high-capacity ships excluded

B. Economic Viability and Profitability

The objective value, representing the sovereignty-adjusted fleet profit, steadily decreases as the sovereignty requirement intensifies. At $\theta = 0.25$ and $\theta = 0.50$, the model yields high optimal values of respectively 8.67 and 8.63 million units. At $\theta = 0.88$, the profit collapses to just 62.8k units due to strict filtering of eligible vessels. Fig. 2 presents the evolution of optimal objective value by sovereignty threshold.

C. Environmental Factor

Table VII and Table VIII presents the top ten most carbon-efficient vessel-route pairings across the four sovereignty scenarios, ranked by the lowest carbon penalty per TEU transported. A consistent trend emerges: vessels with lower emission coefficients (e_{ir}) are consistently allocated to high-volume corridors, regardless of the sovereignty constraint level θ. When the sovereignty threshold is low or moderate ($\theta = 0.25$, $\theta = 0.50$, $\theta = 0.75$), the allocation tends towards vessels with the best criteria in terms of cost, emissions, and sovereignty. The pool of eligible vessels, however, drastically shrinks when $\theta = 0.88$. Consequently, the model's capacity to sustain environmentally optimal decisions is weakened.

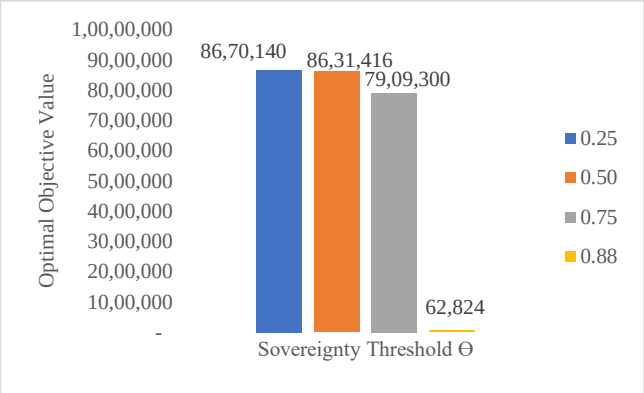


Fig. 2. Evolution of optimal objective value by sovereignty threshold.

TABLE VII. TOP TEN CARBON EFFICIENT PAIRING PER θ

$\theta = 0.25, \theta = 0.50, \theta = 0.75$				
Vessel Id	***Route ID***	***Vessel Route pair***	q_{ir}***value***	e_{ir}
v82	route187	v82route187	120	0,668
v15	route21	v15route21	120	0,668
v18	route66	v18route66	120	0,672
v51	route15	v51route15	120	0,674
v49	route156	v49route156	120	0,678
v1	route187	v1route187	120	0,68
v58	route180	v58route180	120	0,683
v14	route12	v14route12	120	0,683
v76	route56	v76route56	180	1,003
v64	route67	v64route67	180	1,029

TABLE VIII. TOP TEN CARBON EFFICIENT PAIRING PER $\theta = 0.88$

$\theta = 0.88$				
Vessel Id	***Route ID***	***Vessel Route pair***	q_{ir}***value***	e_{ir}
v86	route125	v86route125	15	0,223
v72	route75	v72route75	15	0,234
v92	route37	v92route37	15	0,256
v11	route145	v11route145	15	0,3
v33	route19	v33route19	15	0,435
v94	route68	v94route68	15	0,437
v35	route44	v35route44	15	0,52
v93	route77	v93route77	15	0,56
v8	route114	v8route114	15	0,561
v71	route90	v71route90	120	2,256

D. Resilience and Demand Satisfaction

Resilience, interpreted here through transport demand coverage and unmet demand penalties α_r, exhibits a clear decline when $\theta = 0.88$. For $\theta = 0.25$, $\theta = 0.50$ and $\theta = 0.75$, almost all routes are served, with over 95 per cent of demand satisfied, as shown in Table VII and Table VIII. When $\theta = 0.88$, only a fraction of strategic routes are covered, and small-capacity vessels are unable to absorb large volumes, triggering unmet demand costs.

VI. DISCUSSION

Besides the numerical results, special consideration must be paid to the realization of policy aspects emerging from each scenario. These insights are essential for decision-makers, who must balance short-term financial constraints with the long-term goals of maritime sovereignty and sustainable fleet development.

The analysis highlights that strategic trade-offs become visible through the activation patterns of the fleet. At lower sovereignty thresholds, the solutions consistently point towards compact and efficient fleet structures, largely composed of mid-sized vessels such as Feeders, Medium Containerships, and Small Handies. These vessel types have moderate purchase prices, viable carbon footprints and strong national ownership ratings, making them well-aligned with economic viability and sovereignty objectives. When sovereignty requirements are raised, a marked contraction of fleet diversity is observed, with significant reductions in activation volumes, especially on high-demand routes. Comparable patterns are observed in recent models of oligopolistic 3PL competition, where environmental and digital imperatives, such as IoT adoption, directly affect pricing and deployment strategies [52], confirming the central role of sovereignty and sustainability in nowadays logistics network design. Indeed, the exclusion of vessels of type ULCV and Panamax family (Panamax, Post-Panamax and Neo-Panamax) confirms their incompatibility with a sovereign-oriented national fleet due to their high acquisition cost, foreign dominance, and operational scale requirements. This evolution highlights a fundamental trade-off that underlies the nature of strategic fleet planning: the higher maritime sovereignty becomes a central goal, the more the model leans toward limited vessel options. As a matter of fact, these results justify the choice of several developing maritime nations of the cabotage or the Short-Sea Shipping (SSS) as a strategic choice, leaving the long routes and corridors to international firms.

The discussion of activation patterns must also be complemented by the evolution of the fleet size. When $\theta = 0.25$ or $\theta = 0.50$, the model result is respectively a fleet of 17 vessels that are cost-efficient and sovereignty sensitive. Consequently, the profitability is maintained without overextending the fleet thanks to this controlled activation, which follows the logic of economic viability. The number of activated vessels increases significantly to 53, when $\theta = 0.75$, suggesting a diversification strategy that covers a wider range of demand. However, the number of activated vessels decreases to 18 once more when the sovereignty constraint reaches $\theta = 0.88$. Because of the model's stringent national control, which prevents many large-capacity vessels from activating, this represents a strategic narrowing. The trade-off between autonomy and scale is highlighted by this fluctuation in fleet size: operational breadth is given up in order to preserve political and financial control over maritime assets as sovereignty is maximized.

From an economic perspective, the steady reduction in profit with increasing sovereignty thresholds highlights the

measurable financial cost of sovereignty. This degradation, as shown in Fig. 2 , reflects the inherent cost of sovereignty prioritization: limiting access to high-capacity, cost-efficient ships translates into suboptimal network coverage and inflated operational expenses. Moreover, investment costs rise with the number of activated vessels needed to meet coverage requirements under constrained types.

Moderate thresholds preserve profitability, whereas maximalist sovereignty requirements collapse economic performance by excluding cost-efficient, large-capacity vessels. From a policymaking perspective, this analysis helps quantify the economic sacrifices associated with asserting maritime sovereignty, a pivotal trade-off at the core of strategic fleet planning.

The analysis further reveals how sovereignty constraints shape environmental outcomes and carbon efficiency. The results in Table VII and Table VIII show that the optimization algorithm favors vessels with strong environmental performance, especially on high-demand routes. At a lower or moderate sovereignty threshold, the model benefits from greater flexibility in vessel selection. Consequently, a balance is set between minimizing environmental impact and maximizing capacity utilization. At a higher sovereignty threshold, the model still finds a core set of ships that meet national ownership requirements and environmental standards, pointing to the potential for a foundational "sovereign-green" fleet. Similar findings are reported in recent stochastic ship design models, which show that LNG and methanol configurations are relatively robust initial choices because of their retrofit potential towards ammonia or hydrogen, providing flexibility under uncertain fuel and carbon prices [53]. One noteworthy finding is that, when this restriction is rigorously enforced, vessels with low sovereignty ratings but high environmental scores are left out of the solution, underscoring the dominating influence of sovereignty. This highlights a significant obstacle in sustainable fleet planning: if national control goals are involved, carbon efficiency alone is insufficient. In fact, recent studies underline that decarbonization policies will require more than technological improvements, as carbon pricing and alternative fuels are expected to significantly reshape fleet economics [54].

When combined, these results highlight the necessity of well-thought-out investment plans. Consequently, for countries such as Morocco, the government must encourage the retrofitting or local purchase of mid-sized ships that combine acceptable emissions profiles with significant national ownership potential, if it hopes to build a fleet that is both sovereign and environmentally conscious.

These insights may guide targeted green fleet development, especially for routes with high transport demand and strict environmental thresholds.

As per the resilience analysis, captured through demand coverage, the simulation underscores the systemic fragility introduced by overly restrictive sovereignty rules. The simulation highlights a resilience paradox: while sovereign fleets may be politically desirable, they may also struggle to absorb operational shocks or meet rising freight demand, unless complemented by supportive policy instruments or hybrid strategies (e.g., public-private joint ventures).

To validate the consistency of the proposed model, the sovereignty coefficient f_i was neutralized (set to 1 for all vessels), so that the formulation approximates a standard fleet sizing problem. The model's results in this sovereignty-free form are in line with two well-known studies [46] and [16]. The model emphasizes how operational costs and vessel capacity shape fleet composition. While smaller units offer flexibility, larger ships dominate when demand is high and costs are crucial. Thus; it concentrates on strategic fleet composition and environmental penalties, while [46] prioritized sailing speed and detailed routing. Despite this, both models converge toward a similar balance between cost effectiveness, emissions, and fleet diversity. Moreover,[16] who investigated multi-period fleet deployment under demand uncertainty, show also a similar alignment. The model's sovereignty-free version demonstrates that, in the absence of policy restrictions, demand satisfaction and profitability naturally determine the best fleet composition. The main difference is temporal: the current model is single-period and strategic, whereas [16] consider multi-period stochastic dynamics, but the fundamental economic reasoning is the same.

Importantly, only sovereignty was neutralized for this validation. Other parameters, such as resilience penalties and environmental constraints, were preserved because they form an integral part of the model's originality and are not explicitly treated in [16] and [46]. By incorporating sovereignty as a new strategic criterion, the suggested model not only replicates the insights of [46] and [16] but also builds upon them, ensuring that the comparison is both equitable and methodologically sound.

When combined, these results highlight how sovereignty-maximalist methods diminish resilience, flexibility, profitability, and environmental performance, making them insufficient to maintain strategic autonomy on their own. The implications for policymakers are clear: calibrated strategies that balance national control with operational and economic viability are essential. As a result, the model offers a decision-support tool that highlights the risks to resilience and measures the financial and environmental costs of sovereignty.

The limitations of this study must also be acknowledged. Long-term dynamics and uncertainty are simplified by the single-period, deterministic formulation. To improve realism, extensions should include geopolitical risk factors, stochastic trade flows, and multi-period horizons. In order to scale the model to larger problem instances, future research could also investigate heuristic or hybrid solution approaches. As a matter of fact, recent studies emphasize that near-zero emission fleet transitions depend on integrated strategies that jointly address renewal, deployment, and alternative fuel adoption under multiple sources of uncertainty [55]. Notwithstanding these drawbacks, the model already provides valuable information for building autonomous, robust, and sustainable fleets in developing maritime countries.

VII. Conclusion

This study introduces a strategic fleet sizing model tailored for public maritime authorities operating under sovereignty, environmental, demand coverage, and budgetary constraints. Through a sovereignty coefficient, the model directly incorporates national interest into the objective function, in contrast to traditional firm-level optimization techniques. It uses a mixed-integer formulation based on liner shipping dynamics and public policy considerations to capture decision-making at three levels: vessel activation, route assignment, and deployment of transport volumes.

The trade-offs brought about by sovereignty constraints are demonstrated by computational experiments conducted on a medium-scale dataset (100 vessels, 230 routes). The model gives cost-effectiveness and environmental performance priority at lower thresholds. However, the solution space shrinks as sovereignty requirements increase: the fleet is more vulnerable to resilience threats, flexibility decreases, and demand coverage deteriorates. Although a sovereignty-maximalist approach supports the goals of national control, it significantly reduces the network's overall capacity and economic viability.

These results highlight the significance of calibrated policymaking. Rigid constraints alone cannot sustain sovereignty and strategic autonomy. Indeed, a delicate balance between operability and control is needed. For maritime nations like Morocco, the suggested model provides a useful tool for decision-making that helps in resolving the delicate conflict between sovereign imperatives and economic rationality. Subsequent extensions might investigate dynamic, multi-period formulations or add uncertainty by incorporating geopolitical risks and stochastic trade flows. Nevertheless, the model already offers important information for building robust national managed fleets.

Acknowledgment

This research study has been produced by the LASTIMI Laboratory of Mohammed V University in Rabat. A Special thank you goes to the teams of the CELOG for their precious support. A special thank you also goes to two anonymous reviewers for their valuable feedback that added a lot of value to this study.

Declaration of Competing Interests

The authors have no relevant financial or non-financial interests to disclose.

References

[1] G. Kahyarara and D. Simon, “UNCTAD Ad Hoc expert meeting ‘Maritime transport In Africa: challenges, opportunities, and an agenda for future research’ opportunity and growth diagnostic of maritime transportation in the eastern and southern Africa,” Dar-es-Salaam, 2018.

[2] D. Saidi, A. Ait Bassou, J. El Alami, and M. Hlyal, “Sustainability and resilience analysis in supply chain considering pricing policies and government economic measures,” Rabat, 2024. doi: 10.14569/IJACSA.2024.0150127.

[3] Chilaka.E, “Shipping trade in West Africa during pre colonial and colonial times,” 2015.

[4] Iheduru.O.C, “The political economy of international shipping in West Africa,” Connecticut, 1992.

[5] A. J. Baird, “Privatization trends at the world’s top-100 container ports,” Maritime Policy and Management, vol. 29, no. 3, pp. 271–284, 2002, doi: 10.1080/03088830210132579.

[6] M. R. Brooks and A. A. Pallis, “Assessing port governance models: process and performance components,” Maritime Policy and Management, vol. 35, no. 4, pp. 411–432, 2008, doi: 10.1080/03088830802215060.

[7] G. Wilmsmeier and J. Hoffmann, “Liner shipping connectivity and port infrastructure as determinants of freight rates in the Caribbean,” in Maritime Economics and Logistics, Mar. 2008, pp. 130–151. doi: 10.1057/palgrave.mel.9100195.

[8] I. M. Okafor-Yarwood and F. C. Onuoha, “Whose security is it? elitism and the global approach to maritime security in Africa,” Third World Q, vol. 44, no. 5, pp. 946–966, 2023, doi: 10.1080/01436597.2023.2167706.

[9] B. K. Gesami, “Maritime security in Africa: the Africa Union’s challenge in implementing the 2050 AIM strategy,” 2021. [Online]. Available: https://ssrn.com/abstract=3925071

[10] ALG Transportation Infrastructure & Logistics, “Étude sur la stratégie du secteur du transport maritime Marocain et le développement du pavillon national au Maroc,” Rabat, Jun. 2013.

[11] M. Mezene, “The maritime transport in Morocco : the bottom and the form,” Revue Internationale des Sciences de Gestion, 2019, [Online]. Available: www.revue-isg.com

[12] G. Pantuso, K. Fagerholt, and L. M. Hvattum, “A survey on maritime fleet size and mix problems,” Eur J Oper Res, vol. 235, no. 2, pp. 341–349, Jun. 2014, doi: 10.1016/j.ejor.2013.04.058.

[13] J. L. Everett, A. C. Hax, V. A. Lewinson, and D. Nudds, “Optimization of a fleet of large tankers and bulkers: a linear programming approach,” Marine Technology and SNAME News, vol. 9, no. 04, pp. 430–438, Oct. 1972, doi: 10.5957/mt1.1972.9.4.430.

[14] Bradley, “Planning the mission and composition of US merchant marine fleet,” 1977.

[15] X. Wang, K. Fagerholt, and S. W. Wallace, “Planning for charters: a stochastic maritime fleet composition and deployment problem,” Omega (United Kingdom), vol. 79, pp. 54–66, Sep. 2017, doi: 10.1016/j.omega.2017.07.007.

[16] Q. Meng and T. Wang, “A scenario-based dynamic programming model for multi-period liner ship fleet planning,” Transp Res E Logist Transp Rev, vol. 47, no. 4, pp. 401–413, 2011, doi: 10.1016/j.tre.2010.12.005.

[17] K. Hiekata, S. Wanaka, and Y. Okubo, “Mining rules of decision-making for fleet composition under market uncertainty using a genetic algorithm,” Journal of Marine Science and Technology (Japan), vol. 27, no. 1, pp. 730–739, Mar. 2022, doi: 10.1007/s00773-021-00864-4.

[18] S. Ehlers, H. Pache, F. von Bock und Polach, and T. Johnsen, “A fleet efficiency factor for fleet size and mix problems using particle swarm optimisation,” Ship Technology Research, vol. 66, no. 2, pp. 106–116, May 2018, doi: 10.1080/09377255.2018.1558612.

[19] A. Hoff, H. Andersson, M. Christiansen, G. Hasle, and A. Løkketangen, “Industrial aspects and literature survey: fleet composition and routing,” Comput Oper Res, vol. 37, no. 12, pp. 2041–2061, Dec. 2010, doi: 10.1016/j.cor.2010.03.015.

[20] J. F. Alvarez, P. Tsilingiris, E. S. Engebrethsen, and N. M. P. Kakalis, “Robust fleet sizing and deployment for industrial and independent bulk ocean shipping companies,” INFOR, vol. 49, no. 2, pp. 93–107, May 2011, doi: 10.3138/infor.49.2.093.

[21] M. Giovannini and H. N. Psaraftis, “The profit maximizing liner shipping problem with flexible frequencies: logistical and environmental considerations,” Flex Serv Manuf J, vol. 31, no. 3, pp. 567–597, Sep. 2018, doi: 10.1007/s10696-018-9308-z.

[22] G. Pantuso, K. Fagerholt, and S. W. Wallace, “Uncertainty in fleet renewal: a case from maritime transportation,” Transportation Science, vol. 50, no. 2, pp. 390–407, May 2016, doi: 10.1287/trsc.2014.0566.

[23] N. Wijnolst, “Dynamics of national fleet development simulation models for maritime planners in developing countries,” 1978.

[24] S. C. Cho and A. N. Perakis, “Optimal liner fleet routing strategies,” Maritime Policy and Management, vol. 23, no. 3, pp. 249–259, 1996, doi: 10.1080/03088839600000087.

[25] M. Crary, L. K. Nozick, and L. R. Whitaker, "Sizing the US destroyer fleet," 2000, [Online]. Available: www.elsevier.com/locate/dsw

[26] Taylor, "Chartering strategies for shipping companies," 1981.

[27] K. Fagerholt and H. Kon Lindstad, "Optimal policies for maintaining a supply service in the Norwegian Sea," 1999, [Online]. Available: www.elsevier.com/locate/orms

[28] K. Cullinane and M. Khanna, "Economies of scale in large containerships: optimal size and geographical implications," 2000, [Online]. Available: www.elsevier.com/locate/jtrangeo

[29] Bendall, "A scheduling model for a High speed containership service: a hub and spoke Short-Sea application," International Journal of Maritime Economics, vol. 3, pp. 262–277, 2001, [Online]. Available: www.palgrave-journals.com/ijme

[30] K. Fagerholt, "Optimal fleet design in a ship routing problem," 1999, [Online]. Available: www.elsevier.com/locate/orms

[31] C. Vad Karsten, B. D. Brouer, and D. Pisinger, "Competitive liner shipping network design," Comput Oper Res, vol. 87, pp. 125–136, Nov. 2017, doi: 10.1016/j.cor.2017.05.018.

[32] Ø. S. Patricksson, K. Fagerholt, and J. G. Rakke, "The fleet renewal problem with regional emission limitations: case study from Roll-on/Roll-off shipping," Transp Res Part C Emerg Technol, vol. 56, pp. 346–358, 2015, doi: 10.1016/j.trc.2015.04.019.

[33] A. Konstantinus and J. Woxenius, "Case study: coastal shipping in sub-Saharan Africa," Case Stud Transp Policy, vol. 10, no. 4, pp. 2064–2074, Dec. 2022, doi: 10.1016/j.cstp.2022.09.008.

[34] A. Oyenuga, "Perspectives on the impact of the COVID-19 pandemic on the global and African maritime transport sectors, and the potential implications for Africa's maritime governance," WMU Journal of Maritime Affairs, vol. 20, no. 2, pp. 215–245, Jun. 2021, doi: 10.1007/s13437-021-00233-3.

[35] M. Hachemane and R. Annane, "Maritime transport and port services in Algeria: assessment and development opportunities within the framework of the blue economy," 2024.

[36] S. Domergue, "Maritime security in the Caribbean: causes and Impacts of the regionalism of small and micro-states," Geopolitics, vol. 30, no. 2, pp. 921–959, 2025, doi: 10.1080/14650045.2024.2407596.

[37] F. von Malmborg, "Tapping the conversation on the meaning of decarbonization: discourses and discursive agency in EU politics on low-carbon fuels for maritime shipping," Sustainability (Switzerland) , vol. 16, no. 13, Jul. 2024, doi: 10.3390/su16135589.

[38] N. Friedman, "The maritime strategy and the design of the U.S. fleet," Comparative Strategy, vol. 6, no. 4, pp. 415–435, 1987, doi: 10.1080/01495938708402723.

[39] P. Zhang and L. Drumm, "The flagging-out strategy: an examination of the impacts on the decreasing German national fleet," Mar Policy, vol. 115, May 2020, doi: 10.1016/j.marpol.2020.103872.

[40] P. M. C. E. Panggabean, E. P. Duarte, H. Tarigan, and K. Prihantoro, "Indonesia's maritime defense strategy for securing North Natuna 2019-2024," Formosa Journal of Multidisciplinary Research, vol. 4, no. 3, pp. 1451–1464, Mar. 2025, doi: 10.55927/fjmr.v4i3.124.

[41] S. B. Lee and Y. Kim, "A study on the relationship between national controlling fleets and the managerial performance of ship management companies in Korea," Port Res, vol. 48, no. 2, pp. 2093–8470, 2024, doi: 10.5394/KINPR.2024.48.2.104.

[42] K. Fagerholt, M. Christiansen, L. Magnus Hvattum, T. A. V. Johnsen, and T. J. Vabø, "A decision support methodology for strategic planning in maritime transportation," Omega (Westport), vol. 38, no. 6, pp. 465–474, 2010, doi: 10.1016/j.omega.2009.12.003.

[43] J. Fonseca Ribeiro, "The greening of maritime transportation, energy and climate infrastructures: role of atlantic port-cities," 2017.

[44] G. Prause, The maritime perspective of Africa. 2024. [Online]. Available: https://www.fww.hs-wismar.de/

[45] L. Li, "Building up a sustainable path to maritime security: an analytical framework and its policy applications," Sustainability (Switzerland), vol. 15, no. 8, Apr. 2023, doi: 10.3390/su15086757.

[46] H. J. Kim, D. H. Son, W. Yang, and J. G. Kim, "Liner ship routing with speed and fleet size optimization," KSCE Journal of Civil Engineering, vol. 23, no. 3, pp. 1341–1350, Mar. 2019, doi: 10.1007/s12205-019-0564-6.

[47] O. Loennechen, K. Fagerholt, B. Lagemann, and M. Stålhane, "Maritime fleet composition under future greenhouse gas emission restrictions and uncertain fuel prices," Maritime Transport Research, vol. 6, Jun. 2024, doi: 10.1016/j.martra.2024.100103.

[48] M. Christiansen, K. Fagerholt, and D. Ronen, "Ship routing and scheduling: status and perspectives," Transportation Science, vol. 38, no. 1, pp. 1–18, 2004, doi: 10.1287/trsc.1030.0036.

[49] P. Cariou, "Liner shipping strategies: an overview," Int. J. Ocean Systems Management, vol. 1, no. 1, pp. 2–13, 2008.

[50] D. Saidi, A. Ait Bassou, M. Hlyal, and J. El Alami, "Analyzing quantity-based strategies for supply chain sustainability and resilience in uncertain environment," Rabat, 2024. doi: 10.14569/IJACSA.2024.0150533.

[51] B. D. Brouer, C. V. Karsten, and D. Pisinger, "Optimization in liner shipping," 4OR, vol. 15, no. 1, pp. 1–35, Mar. 2017, doi: 10.1007/s10288-017-0342-6.

[52] K. Izikki, A. Ait Bassou, M. Hlyal, and J. El Alami, "Impact of the IoT integration and sustainability on competition within an oligopolistic 3PL market," Rabat, 2024. doi: 10.14569/IJACSA.2024.01504107.

[53] B. Lagemann, S. Lagouvardou, E. Lindstad, K. Fagerholt, H. N. Psaraftis, and S. O. Erikstad, "Optimal ship lifetime fuel and power system selection under uncertainty," Transp Res D Transp Environ, vol. 119, Jun. 2023, doi: 10.1016/j.trd.2023.103748.

[54] K. Cullinane and J. Yang, "Evaluating the costs of decarbonizing the shipping industry: a review of the literature," Jul. 01, 2022, MDPI. doi: 10.3390/jmse10070946.

[55] Y. Wang and Ç. Iris, "Transition to near-zero emission shipping fleet powered by alternative fuels under uncertainty," Transp Res D Transp Environ, vol. 142, May 2025, doi: 10.1016/j.trd.2025.104689.

Prediction of Mining-Induced Subsidence in Saudi Arabia Phosphate Mines Using ANN Method

Atef GHARBI[1], Mohamed AYARI[2], Yamen El Touati[3],
Zeineb Klai[4], Mahmoud Salaheldin Elsayed[5], Elsaid Md. Abdelrahim[6]
Department of Information Systems-Faculty of Computing and Information Technology,
Northern Border University, Rafha 91911, Saudi Arabia[1]
Department of Information Technology-Faculty of Computing and Information Technology,
Northern Border University, Rafha 91911, Saudi Arabia[2]
Department of Computer Sciences-Faculty of Computing and Information Technology,
Northern Border University, Rafha, Saudi Arabia[3, 4, 5]
Computer Science Department-Science College, Northern Border University (NBU), Arar 73213, Saudi Arabia[6]

***Abstract*—This study develops and validates an artificial neural network (ANN) model to predict mining-induced land subsidence in Saudi Arabia's Al-Jalamid and Umm Wu'al phosphate mines. A multilayer perceptron is used with optimized hyperparameters based on four inputs (ground point position, distance from extraction center, accumulated exploitation volume, and time). The optimal configuration (5 hidden layers, 64 nodes, 240 epochs) achieves RMSE = 22 mm and MAE = 13 mm, outperforming traditional numerical/statistical baselines. Case-study validation at both mines confirms robustness (e.g., RMSE ≈ 20 mm, MAE ≈ 12 mm), enabling practical mitigation such as ground reinforcement and extraction-rate control. The results demonstrate that a tuned ANN provides accurate, operationally useful subsidence forecasts, supporting safer and more sustainable mine planning.**

Keywords*—Subsidence prediction; phosphate mine; artificial neural network; multilayer perceptron; hyperparameter optimization

I. Introduction

Phosphate mining is a cornerstone of Saudi Arabia's mining industry, playing a pivotal role in the nation's economic development. As one of the world's largest producers of phosphate, Saudi Arabia contributes significantly to global fertilizer production, which is essential for sustaining agricultural productivity and food security. The Al-Jalamid and Umm Wu'al phosphate mines, located in the Northern Borders Province, are among the most significant reserves globally, with extensive deposits that support both domestic and international demand. However, the extraction of phosphate is not without its challenges. Mining activities, particularly underground operations, often lead to environmental and geological disturbances, with land subsidence being one of the most critical issues. Subsidence can result in the sinking or settling of the ground surface, leading to infrastructure damage, environmental degradation, and increased operational risks. These impacts are exacerbated in arid regions like Saudi Arabia, where the geological formations are susceptible to deformation due to mining activities.

Mining-induced subsidence poses significant risks to both the environment and infrastructure. In Saudi Arabian phosphate mines, the extraction of phosphate-rich layers disrupts the geological equilibrium, causing the overlying strata to collapse or settle. This subsidence can lead to the formation of sinkholes, cracks in the ground surface, and damage to nearby infrastructure such as roads, pipelines, and buildings. Additionally, subsidence can alter groundwater flow patterns, further exacerbating environmental concerns in a region already facing water scarcity. Accurate prediction of subsidence is therefore essential for mitigating these risks, enabling mine operators to implement preventive measures, optimize extraction processes, and ensure the safety and sustainability of mining operations. Despite advancements in predictive modeling, the complex interplay of geological, hydrological, and operational factors makes subsidence prediction a challenging task, necessitating the development of robust and reliable models.

Over the past decade, various methodologies have been proposed to predict mining-induced subsidence, ranging from traditional numerical: [1] analyzed land subsidence using monitoring and numerical simulation in linear engineering areas, [2] modeled surface subsidence in coal mines through a bonded block numerical method, and [3] examined the geotechnical characterization of phosphate mining waste materials for pavement applications. Furthermore, [4] compared statistical and machine learning approaches for land subsidence modelling, [5] proposed a novel prediction method using data assimilation techniques, and [6] introduced a probability integral model integrated with active and passive remote sensing data to monitor subsidence. Numerical methods, such as the two-dimensional seepage model, quasi-three-dimensional seepage model, and fully coupled three-dimensional method, offer relatively straightforward implementation and interpretation [7-8]. While these methods provide detailed insights into subsidence mechanisms, they often require extensive computational resources and precise input parameters, limiting their practical applicability.

Statistical techniques, including time-series modeling [9-11], regression analysis [12-13], and Grey theory [14-16] offer simpler alternatives but are often constrained by their inability to capture the nonlinear relationships inherent in subsidence data.

Recent advancements in artificial intelligence (AI) have revolutionized subsidence prediction, with machine learning models such as Support Vector Machines (SVMs) and Artificial Neural Networks (ANNs) demonstrating superior performance. SVMs, known for their ability to handle high-dimensional data, have been successfully applied in subsidence prediction, particularly in cases with limited datasets [17-18]. However, their scalability remains a concern for large-scale mining operations. ANNs, on the other hand, have gained prominence due to their ability to model complex, nonlinear relationships and adapt to diverse datasets [19-21]. Studies have highlighted the effectiveness of ANNs in predicting subsidence, particularly when integrated with remote sensing and geographic information system (GIS) data. Despite these advancements, there is a need for further research to optimize ANN architectures, improve prediction accuracy, and address the challenges associated with long-term subsidence forecasting. The ANN model can be used to predict subsidence at different locations and times, helping mine operators to minimize subsidence, implement mitigation measures to protect infrastructure and the environment, monitor subsidence trends over time.

Despite various numerical and statistical models for subsidence prediction, their applicability is often limited by computational constraints and the complexity of subsurface interactions. Traditional methods struggle to account for nonlinear geological behaviors, making them unsuitable for large-scale, real-time applications. This study aims to address this gap by developing an ANN model tailored to the unique geological and operational conditions of Saudi Arabian phosphate mines offering a robust alternative for subsidence prediction.

By leveraging advanced machine learning techniques, the proposed model seeks to provide accurate and reliable subsidence predictions, contributing to the sustainable management of mining operations and the mitigation of environmental risks.

This study presents an artificial neural network (ANN) model for predicting mining-induced subsidence in Saudi Arabian phosphate mines. The paper is structured as follows: Section II provides an overview of the study area, including geological characteristics and mining conditions. Section III details the methodology, describing the ANN model architecture, feature selection, and training process. Section IV presents the results, including model performance evaluation and validation. Finally, Section V concludes the study with key insights and recommendations for future research.

II. Study Area

As two of the world's largest phosphate reserves, the Al-Jalamid and Umm Wu'al mines in Saudi Arabia's Northern Borders Province possess unique geological and economic significance. Their sedimentary formations, characterized by alternating limestone, shale, and phosphorite layers, make them highly susceptible to subsidence. Previous studies have reported significant subsidence events in these mines, highlighting the need for advanced predictive models. The presence of Tunnel Boring Machines (TBMs) further complicates ground stability, making these sites ideal for testing AI-driven predictive frameworks.

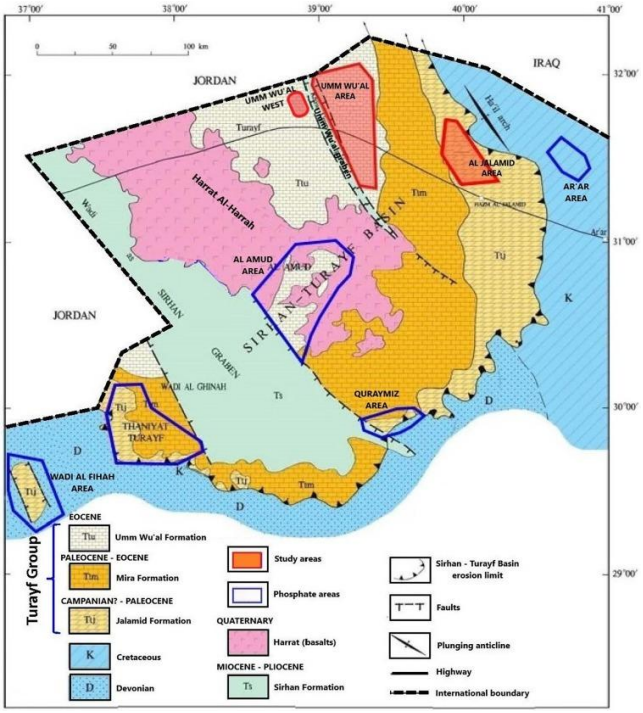


Fig. 1. Geological framework of the sirhan-turayf basin: distribution of phosphate deposits and study sites.

Fig. 1 presents the Al-Jalamid and Umm Wu'al phosphate mines in northern Saudi Arabia, among the world's largest phosphate reserves, crucial for global fertilizer production. Al-Jalamid is a major underground mining site where extraction has led to land subsidence and high disc cutter consumption in TBMs. The region's sedimentary formations (limestone, shale, phosphate-rich layers) present geological and environmental challenges due to its arid climate and limited groundwater.

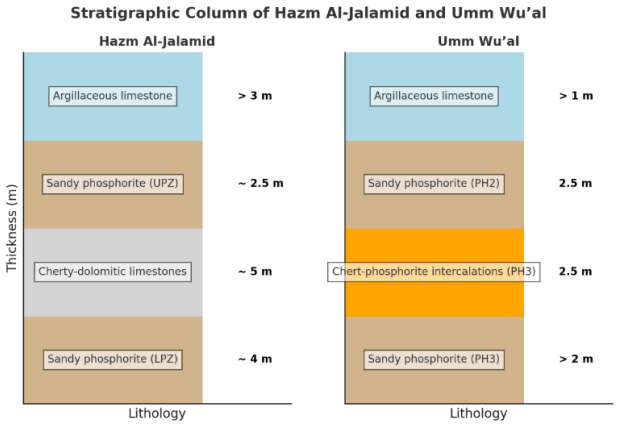


Fig. 2. Stratigraphic columns of phosphate-bearing formations in Hazm Al-Jalamid and Umm Wu'al.

Fig. 2 introduces the phosphorite deposits at Hazm Al-Jalamid range from friable sandy-calcareous to hard compact formations, reflecting depositional and diagenetic variations that influence mining methods and geotechnical behavior. The

Thaniyat Phosphorite Member comprises two main phosphorite horizons: Lower Phosphorite Zone (LPZ) and Upper Phosphorite Zone (UPZ).

The LPZ consists of calcareous and dolomitic phosphorite beds (3–4 m thick), rich in bioclastic fragments, indicating a marine depositional environment. The UPZ, 4–7 m thick (avg. 6 m), contains sparry calcite lenses, argillaceous cement, and vuggy structures, suggesting post-depositional diagenesis and bioturbation. A 12-meter-thick sand, clay, and limestone sequence overlies the UPZ.

An intermediate horizon (0.5–5 m thick) between LPZ and UPZ consists of low-grade phosphatic dolomitic limestone with quartz geodes and chert nodules, marking a transitional depositional setting influenced by sedimentological and chemical variations.

III. Artificial Neural Networks

The dataset employed in this study was collected from monitoring records of the Al-Jalamid and Umm Wu'al phosphate mines located in the Northern Borders Province of Saudi Arabia. It consists of ground deformation measurements covering multiple observation points distributed across subsidence-prone areas. Four primary variables were included: (i) ground point positions (Y), representing the spatial coordinates of monitoring stations along the subsidence trough; (ii) distance from the extraction chamber center (L), indicating the horizontal offset of each point from the center of mining activity; (iii) accumulated exploitation volume (V), defined as the total phosphate volume extracted up to a given measurement epoch; and (iv) time (T), denoting the elapsed duration since the initiation of mining operations.

The dataset underwent several preprocessing steps to ensure consistency and reliability. Missing values were handled using linear interpolation for continuous features, while outliers were identified and removed. All input features were normalized to a [0, 1] range to eliminate scale differences and enhance model convergence. Following preprocessing, the dataset was divided into three subsets: 70% for training the artificial neural network (ANN), 15% for hyperparameter validation, and 15% for final performance testing. This partitioning ensured robust evaluation and reduced overfitting risks.

A. Artificial Neural Network Architecture

Fig. 3 presents the proposed ANN model is designed to predict mining-induced subsidence using a multilayer perceptron (MLP) architecture. The model consists of three primary components: an input layer, hidden layers, and an output layer.

1) Input layer: The input layer comprises four neurons, each corresponding to one of the input variables:

a) Ground point positions (Y): The spatial coordinates of monitoring points along the subsidence trough.

b) Distance from extraction chamber center (L): The horizontal distance from the center of the mining activity to each monitoring point.

c) Accumulated exploitation volume (V): The total volume of phosphate extracted up to the measurement epoch.

d) Time (T): The elapsed time since the start of mining operations.

2) Hidden layers: The model includes two hidden layers, each containing 64 neurons. The number of hidden layers and neurons was determined through validation, balancing model complexity and generalization performance. The Rectified Linear Unit (ReLU) activation function was used for the hidden layers, as it effectively mitigates the vanishing gradient problem and accelerates convergence during training.

3) Output layer: The output layer consists of a single neuron that predicts the subsidence value (η). A sigmoid activation function was employed in the output layer to ensure that the predicted subsidence values fall within a realistic range (0 to 1, normalized to the maximum observed subsidence).

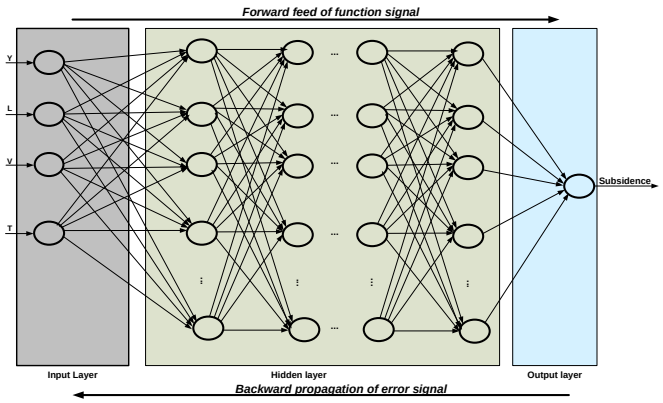


Fig. 3. Architecture of the Backpropagation Neural Network (BPNN) for subsidence prediction.

B. Data Preprocessing

The dataset used for training and testing the ANN model was preprocessed to ensure optimal performance. The preprocessing steps included as it is illustrated in Fig. 4:

1) Data cleaning: Missing values in the dataset were addressed using linear interpolation for continuous variables and mode imputation for categorical variables. Outliers were detected and eliminated from the dataset.

2) Normalization: All input variables were normalized to a range of [0, 1] using min-max scaling to ensure that features with larger magnitudes did not dominate the training process. The normalization formula is given by:

$$X_{norm}=(X-X_{min})/(X_{max}-X_{min})$$

where X_{min} and X_{max} are the minimum and maximum values of the feature, respectively.

3) Feature selection: A correlation analysis was conducted to identify and remove highly correlated features, reducing redundancy and improving model efficiency. The final set of features included ground point positions, distance

from the extraction chamber center, accumulated exploitation volume, and time.

4) Data splitting: To ensure rigorous model assessment, the data were split into three subsets: 70% for training the ANN, 15% for hyperparameter validation, and 15% for final performance testing.

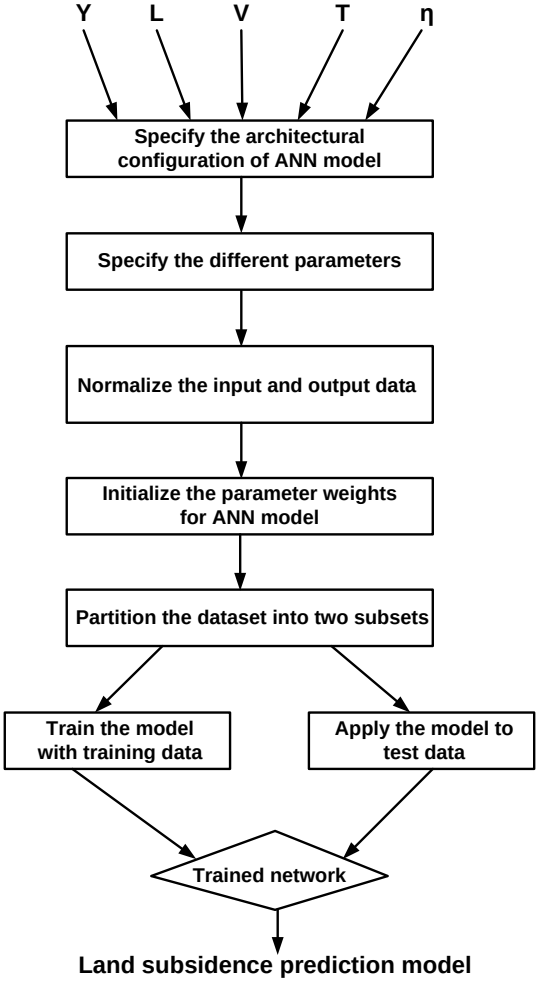


Fig. 4. Workflow of the ANN model for land subsidence prediction.

C. Model Training

In Artificial Neural Network (ANN) models, each neuron in the input layer receives an assigned input, which is transmitted to neurons in subsequent layers through weighted connections. These weights, ranging from -1 to 1, represent the relative importance of each connection. To determine the neuron's response, an activation function is applied to the weighted input, producing an output value. If this value exceeds a predefined threshold, the neuron becomes active; otherwise, it remains inactive.

In the output layer, the discrepancy between the predicted and actual output is quantified as an error. Through a process known as backpropagation, this error propagates backward through the network, adjusting the weights of preceding layers. This iterative process continues until the error converges below a specified threshold, optimizing the network's predictive accuracy.

Following the training phase, the validation and testing phases commence. During validation, hyperparameters such as training duration, learning rates, and the number of neurons in hidden layers are fine-tuned to enhance model performance. The testing phase assesses the model's generalization ability, evaluating its predictive accuracy on unseen data. The final model is selected based on its optimal performance metrics.

The GIS-integrated ANN framework utilized a backpropagation neural network (BPNN) for land subsidence susceptibility assessment, with performance evaluated via correlation coefficient. The training phase mathematics and convergence behavior are derived as follows:

The outputs of the hidden layer h_i and output layer O_k are given by Eq. (1) and Eq. (2).

$$h_i = F(V_j) = \left[B_{oj} + \sum_{i=1}^{I} (B_{ij} \cdot v_i) \right] \quad (1)$$

$$O_k = F(H_k) = \left[B_{ok} + \sum_{j=1}^{J} (B_{kj} \cdot h_i) \right] \quad (2)$$

where V_j and H_k denote the intermediate computations before activation; B_{oj} and B_{ok} correspond to bias terms that establish threshold values; F signifies the activation function operating within a range of 0 to 1; and v_i, h_i, and O_k refer to the input, hidden, and output layers, respectively.

For activation functions, the hyperbolic tangent sigmoid is used, which computes the outputs of hidden and output layers as given in Eq. (3) and Eq. (4).

$$h_i = F(V_j) = \frac{1}{1+e^{-V_j}} \quad (3)$$

$$O_k = F(H_k) = \frac{1}{1+e^{-H_k}} \quad (4)$$

The error function E is formulated as shown in Eq. (5).

$$E = \frac{1}{2} \sum_{k=1}^{K} O_k^2 = \frac{1}{2} \sum_{k=1}^{K} (d_k - O_k)^2 \quad (5)$$

Here, E represents the cumulative error across all output nodes, calculated as the sum of squared deviations between the network's predictions and target values. For each output node k, d_k denotes the desired target value, while ε_k corresponds to the prediction error (i.e., $\varepsilon_k = d_k - O_k$,). The weight modification between the hidden and output layers is defined as shown in Eq. (6).

$$\Delta b_{jk} = \alpha \cdot h_j \cdot \delta_k \quad (6)$$

where α is the learning rate. Reformulating Eq. (6) yields Eq. (7).

$$b_{jk}(n+1) = b_{jk}(n) + \Delta b_{jk}(n) \quad (7)$$

where n represents the iteration index. Taking the partial derivative of the error E with respect to b_{ij}, the following relations are derived:

The correlation coefficient (C) served as the primary performance metric for evaluating the GIS-integrated ANN model, it is calculated as shown in Eq. (8).

$$C=\frac{\sum_{i=1}^{n}(d_i-\bar{d})(p_i-\bar{p})}{\sqrt{\sum_{i=1}^{n}(d_i-\bar{d})^2\cdot\sum_{i=1}^{n}(p_i-\bar{p})^2}} \quad (8)$$

where C denotes the correlation coefficient, d_i is the target output, p_i is the predicted output, $\bar{d}$ is the mean target value, and $\bar{p}$ is the mean predicted output. Fig. 3 illustrates the architecture of the proposed predictive framework, which integrates a backpropagation neural network (BPNN) with four critical geospatial input parameters: (i) Percentage of agricultural land use, (ii) Energy consumption by groundwater extraction wells, (iii) Proportion of fine-grained sediments, and (iv) Mean maximum drainage path length.

Fig. 3 presents a comprehensive view of the primary stages in ANN configuration. While defining the network architecture, selecting activation functions, and determining the number of training epochs necessitate manual adjustments, automated processes manage data preprocessing, normalization, and weight initialization.

IV. Model Performance Evaluation

A. Performance Metrics

Multiple combinations of hyperparameters were tested, including:

- Number of hidden layers (e.g., 3, 4, 5, 6, …).
- Number of nodes per hidden layer (e.g., 32, 64, 128, …).
- Number of epochs (e.g., 100, 200, 240, 300, …).

Instead of using a grid search (testing all possible combinations), a random search is used (testing random combinations) to find the optimal hyperparameters. This approach was selected for the following reason. Random search explores the hyperparameter space more efficiently by sampling parameter combinations randomly, rather than exhaustively evaluating all possible combinations as in grid search. This is particularly advantageous when dealing with computationally expensive models.

For each combination of hyperparameters, the RMSE and MAE are recorded on the validation set and calculate the average RMSE and MAE for each hyperparameter combination.

The performance of the models was evaluated using:

- RMSE: Measures the average magnitude of the prediction errors.
- MAE: Provides a robust measure of error, less sensitive to outliers.

B. Optimal Hyperparameters

Table I presents a detailed summary of performance metrics corresponding to various artificial neural network (ANN) architectures. The "Hidden Layers" column denotes the number of hidden layers employed in each architecture, which directly affects the model's capacity to capture non-linear and complex data patterns. The "Hidden Nodes" column indicates the number of neurons per hidden layer; increasing this number generally enhances the model's representational power but may also heighten the risk of overfitting. The "Epochs" column specifies the total number of training iterations over the dataset, where a higher count may improve convergence but similarly raise overfitting concerns. The "Avg. RMSE (mm)" and "Avg. MAE (mm)" columns report the average root mean square error and mean absolute error, respectively, across validation folds—lower values in both metrics are indicative of better predictive performance. Finally, the "Std. Dev. RMSE" and "Std. Dev. MAE" columns reflect the standard deviations of RMSE and MAE, providing insight into the stability and consistency of each model's performance across k-fold cross-validation.

Insights from Table I reveal that as the number of hidden layers and nodes increases, the average RMSE and MAE generally decrease, suggesting that more complex architectures may yield better performance.

TABLE I. PERFORMANCE OF DIFFERENT ANN ARCHITECTURES

Hidden Layers	Hidden Nodes	Epochs	Avg. RMSE (mm)	Avg. MAE (mm)	Std. Dev. RMSE	Std. Dev. MAE
3	32	100	25.3	15.2	1.2	0.8
3	64	100	24.8	14.9	1.1	0.7
3	128	100	24.5	14.7	1.3	0.9
4	32	200	23.8	14.3	1	0.6
4	64	200	23.1	14	0.9	0.5
4	128	200	23.5	14.2	1.1	0.7
5	32	240	22.5	13.5	0.8	0.5
5	**64**	**240**	**22**	**13**	**0.7**	**0.4**
5	128	240	22.3	13.2	0.9	0.6
6	32	300	23	14.1	1	0.6
6	64	300	22.8	13.8	0.9	0.5
6	128	300	23.2	14	1.1	0.7

The bolded row (5 hidden layers, 64 hidden nodes, 240 epochs) represents the optimal configuration, as it achieves the lowest RMSE (22.0 mm) and lowest MAE (13.0 mm) with the smallest standard deviations (0.7 for RMSE and 0.4 for MAE). This configuration strikes a balance between model complexity and generalization, avoiding overfitting while maintaining high predictive accuracy.

Increasing the number of hidden layers beyond 5 does not significantly improve performance and may lead to overfitting, as seen in the 6-layer configurations. A moderate number of

hidden nodes (e.g., 64) provides the best balance between model complexity and performance, with fewer nodes (e.g., 32) resulting in higher errors and more nodes (e.g., 128) not yielding significant improvements. Training for 240 epochs is sufficient for convergence, as extending training to 300 epochs does not improve performance and may increase the risk of overfitting.

C. Case Studies

The ANN model was applied to two real-world case studies in Saudi Arabian phosphate mines to demonstrate its practical applicability:

1) Al-Jalamid mine: The model was used to predict subsidence in the Al-Jalamid mine, where extensive underground mining has led to significant ground deformation. The predictions were compared with actual subsidence measurements, achieving an RMSE of 20 mm and an MAE of 12 mm. The model's accuracy enabled mine operators to implement targeted mitigation measures, such as ground reinforcement and controlled extraction rates, reducing the risk of infrastructure damage.

2) Umm Wu'al mine: In the Umm Wu'al mine, the model was employed to forecast subsidence trends. The predictions were used to optimize mining operations, ensuring that subsidence remained within acceptable limits. The model's ability to provide long-term forecasts was particularly valuable for strategic planning and risk management.

V. Conclusions

This study developed an Artificial Neural Network (ANN) model to predict mining-induced subsidence in Saudi Arabian phosphate mines. The model demonstrated high predictive accuracy, achieving an RMSE of 22 mm, and an MAE of 13 mm, on the testing dataset. Key factors influencing subsidence, including distance from the extraction chamber center and accumulated exploitation volume, were identified through feature importance analysis. The model's robustness and adaptability make it a reliable tool for subsidence prediction in mining operations.

The proposed Artificial Neural Network (ANN) model offers significant practical benefits for the mining industry and environmental management. Risk mitigation is achieved through accurate subsidence prediction, allowing mine operators to implement targeted measures such as ground reinforcement and controlled extraction rates, thereby reducing the risk of infrastructure damage and environmental degradation. The model also enhances operational optimization by forecasting subsidence trends over time, ensuring mining activities remain within safe and sustainable limits. Additionally, it aids in strategic planning by providing long-term subsidence predictions, enabling mining companies to balance economic objectives with environmental and safety considerations. Furthermore, the model promotes environmental protection by minimizing subsidence-related impacts such as groundwater disruption and land deformation, which is particularly crucial in ecologically sensitive regions like Saudi Arabia.

Building upon the current findings, future research will aim to address the identified limitations and enhance the robustness and applicability of the ANN model. First, efforts will focus on integrating higher-resolution and more diverse ground truth datasets to reduce spatial bias and improve model generalization across heterogeneous regions. Second, to improve model interpretability, future studies will explore hybrid approaches that couple ANN models with explainable artificial intelligence techniques, allowing for better attribution of predicted subsidence to underlying physical processes. Finally, transfer learning and domain adaptation strategies will be investigated to facilitate the application of trained models to new geographic areas with differing geologic and anthropogenic characteristics

Acknowledgment

The authors extend their appreciation to the Deanship of Scientific Research at Northern Border University, Arar, KSA for funding this research work through the project number “NBU-FFMRA-2025-2441-01”.

References

[1] C. Jia, X. Yang, J. Wu, P. Ding, and C. Bian, "Monitoring analysis and numerical simulation of the land subsidence in linear engineering areas," KSCE Journal of Civil Engineering, vol. 25, no. 7, Jul. 2021, pp. 2674–2689.

[2] X. Liu, Y. Zhang, J. Zhang, T. Yang, P. Jia, and R. Guo, "Modelling surface subsidence of coal mines using a bonded block numerical method," Geomatics, Natural Hazards and Risk, vol. 15, no. 1, 2024, Art. no. 2336017.

[3] R. Malaoui, M. R. Soltani, A. Djellali, A. Soukeur, and R. Kechiched, "Geotechnical characterization of phosphate mining waste materials for use in pavement construction," Engineering, Technology & Applied Science Research, vol. 13, no. 1, Feb. 2023, pp. 10005–10013.

[4] E. Rafiei Sardooi, H. R. Pourghasemi, A. Azareh, F. Soleimani Sardoo, and J. J. Clague, "Comparison of statistical and machine learning approaches in land subsidence modelling," Geocarto International, vol. 37, no. 21, 2022, pp. 6165–6185.

[5] L. Gazzola et al., "A novel methodological approach for land subsidence prediction through data assimilation techniques," Computational Geosciences, vol. 25, 2021, pp. 1731–1750.

[6] R. Wang et al., "A novel method of monitoring surface subsidence law based on probability integral model combined with active and passive remote sensing data," Remote Sensing, vol. 14, no. 2, Jan. 2022, Art. no. 299.

[7] D. M. Zhang, X. L. Zhang, and W. W. Du, "Discrete element method based investigation on displacement and bearing characteristics of pile foundation under seepage erosion," Rock and Soil Mechanics, vol. 45, no. 4, Apr. 2024, Art. no. 8.

[8] C. Y. Ku and C. Y. Liu, "Modeling of land subsidence using GIS-based artificial neural network in Yunlin County, Taiwan," Scientific Reports, vol. 13, no. 1, Mar. 2023, Art. no. 4090.

[9] A. Aditiya and T. Ito, "Present-day land subsidence over Semarang revealed by time series InSAR new small baseline subset technique," International Journal of Applied Earth Observation and Geoinformation, vol. 125, 2023, Art. no. 103579.

[10] W. L. Hakim et al., "InSAR time-series analysis and susceptibility mapping for land subsidence in Semarang, Indonesia using convolutional neural network and support vector regression," Remote Sensing of Environment, vol. 287, 2023, Art. no. 113453.

[11] G. Wang et al., "Coastal subsidence detection and characterization caused by brine mining over the Yellow River Delta using time series InSAR and PCA," International Journal of Applied Earth Observation and Geoinformation, vol. 114, 2022, Art. no. 103077.

[12] T. Y. Kwak, S. Hong, J. H. Lee, and S. I. Woo, "Analysis of the limitations of the existing subsidence prediction method based on the subsidence measurement data and suggestions for improvement method through weighted nonlinear regression analysis," Journal of the Korean Geotechnical Society, vol. 38, no. 12, 2022, pp. 103–112.

[13] J. Wang et al., "Control and prevent land subsidence caused by foundation pit dewatering in a coastal lowland megacity: indicator definition, numerical simulation, and regression analysis," Environmental Earth Sciences, vol. 82, no. 2, Feb. 2023, Art. no. 66.

[14] J. Li, Z. He, C. Piao, W. Chi, and Y. Lu, "Research on subsidence prediction method of water-conducting fracture zone of overlying strata in coal mine based on grey theory model," Water, vol. 15, no. 23, Dec. 2023, Art. no. 4177.

[15] D. Yuan et al., "Application of optimized grey-Markov model to land subsidence monitoring with InSAR," IEEE Access, vol. 10, 2022, pp. 96720–96730.

[16] B. Li, N. Liu, and W. Wang, "Coupling model based on grey relational analysis and stepwise discriminant analysis for subsidence discrimination of foundation in soft clay areas," International Journal of Computational Science and Engineering, vol. 24, no. 1, 2021, pp. 55–63.

[17] S. Mehrnoor et al., "Land subsidence hazard assessment based on novel hybrid approach: BWM, weighted overlay index (WOI), and support vector machine (SVM)," Natural Hazards, vol. 115, no. 3, 2023, pp. 1997–2030.

[18] L. Wang et al., "Automatic-detection method for mining subsidence basins based on InSAR and CNN-AFSA-SVM," Sustainability, vol. 14, no. 21, Nov. 2022, Art. no. 13898.

[19] D. Zhou, X. Zuo, and Z. Zhao, "Constructing a large-scale urban land subsidence prediction method based on neural network algorithm from the perspective of multiple factors," Remote Sensing, vol. 14, no. 8, Apr. 2022, Art. no. 1803.

[20] S. Kumar, D. Kumar, P. K. Donta, and T. Amgoth, "Land subsidence prediction using recurrent neural networks," Stochastic Environmental Research and Risk Assessment, vol. 36, no. 2, 2022, pp. 373–388.

[21] Q. Qi et al., "Spatial prediction of soil organic carbon in coal mining subsidence areas based on RBF neural network," International Journal of Coal Science & Technology, vol. 10, no. 1, 2023, Art. no. 30.

A Review on Image-Based Methods for Plant Disease Identification in Diverse Data Conditions

Feilong Tang[1], Rosalyn R Porle[2]*, Hoe Tung Yew[3], Farrah Wong[4]
College of Mechanical and Electrical Engineering, Xichang University, Xichang 615013, China[1]
Faculty of Engineering, Universiti Malaysia Sabah, Kota Kinabalu, Sabah, Malaysia[1, 2, 3, 4]

***Abstract*—Image-based plant disease identification methods have demonstrated potential in enhancing crop protection through early detection. However, the development of this field faces several challenges, such as the scarcity of high-quality annotated data, significant intra-class variation and high inter-class similarity among plant diseases, and the limited generalization ability of current models under diverse domain conditions. We extensively investigated 110+ latest papers on plant disease identification, aiming to present a timely and comprehensive overview of the most recent advances in the field, along with impartial comparisons of strengths and weaknesses of the existing works. Specifically, we begin by reviewing traditional machine learning and deep learning methods, which form the foundation for many current models. We then introduce a taxonomy of transfer learning methods, including instance-based, mapping-based, and network-based methods, and analyze their effectiveness in enhancing classification performance by leveraging prior knowledge under data-constrained scenarios. Subsequently, we examine recent advances in few-shot learning methods for plant disease identification, categorizing them into model-based, metric-based, and optimization-based methods, and evaluate their capabilities in addressing data scarcity and improving identification accuracy. Finally, we summarize the current limitations and outline promising future research directions, with the aim of guiding continued development in this area.**

Keywords*—*Few-shot learning; transfer learning; deep learning; crop protection; early detection; data scarcity

I. Introduction

Plant disease identification has emerged as a significant area of research. Timely identification of plant diseases is critical for reducing crop losses and minimizing the use of pesticides [1]. Image-based plant disease identification has emerged as a fast, accurate, and reliable method for early detection [2],[3]. These methods typically involve several stages, such as image acquisition, image processing, feature extraction, and classification.

However, image-based methods face critical challenges. Variations in noise, background, and lighting in plant disease datasets often affect identification accuracy [4]. In addition, the diversity of tasks, such as identifying citrus diseases [5], tomato diseases [6], and wheat diseases [7], combined with the presence of grafted, hybrid, and genetically modified plants, exacerbates issues of data scarcity and class imbalance in plant disease identification. These challenges underscore the need for new image-based techniques, such as few-shot learning (FSL) [8], [9], transfer learning (TL) [10], [11].

Previous studies have explored various methods for plant disease identification. For example, support vector machines (SVMs) [12], convolutional neural networks (CNNs) [13], TL[14], and generative adversarial networks (GANs) [15]. However, existing surveys are predominantly architecture-centric [3],[16], offering limited guidance when data conditions, rather than model choice, are the primary bottleneck. In practice, field performance is more often constrained by domain shift, class imbalance, and scarce labels than by marginal architectural differences. To ground our review, we conducted a structured search across IEEE Xplore, Scopus, Web of Science, and Google Scholar covering 2019 to 2025, complemented by backward and forward citation tracing to capture seminal earlier work. The search queries combined keywords including plant disease, early detection, leaf disease identification, transfer learning, domain adaptation, and few-shot learning. From over 230 retrieved studies, we retained 116 directly relevant papers on image-based plant disease identification.

To address the identified gap, we adopt a scenario-driven taxonomy organized by three data regimes: (i) tasks with sufficient samples, (ii) tasks with limited samples but abundant related task data, and (iii) tasks with limited samples and no related data. The objective is to map method families to each regime. We synthesize prior work, identify recurring limitations, and provide scenario-specific recommendations for method selection under realistic field conditions. For navigation, an overview diagram is placed at the beginning of Section II (Fig. 1), linking data regimes to the technique families reviewed in Sections II-IV.

II. Machine Learning or Deep Learning for Plant Disease Identification

When a sufficient amount of data is available, both machine learning (ML) and deep learning (DL) methods have proven effective in various plant disease classification tasks. In this section, a classification method is proposed that combines traditional ML, DL, and their fusion. Fig. 1 illustrates the classification methods reviewed in this study, providing practical guidance for plant disease identification under varying data scenarios.

*Corresponding Author.

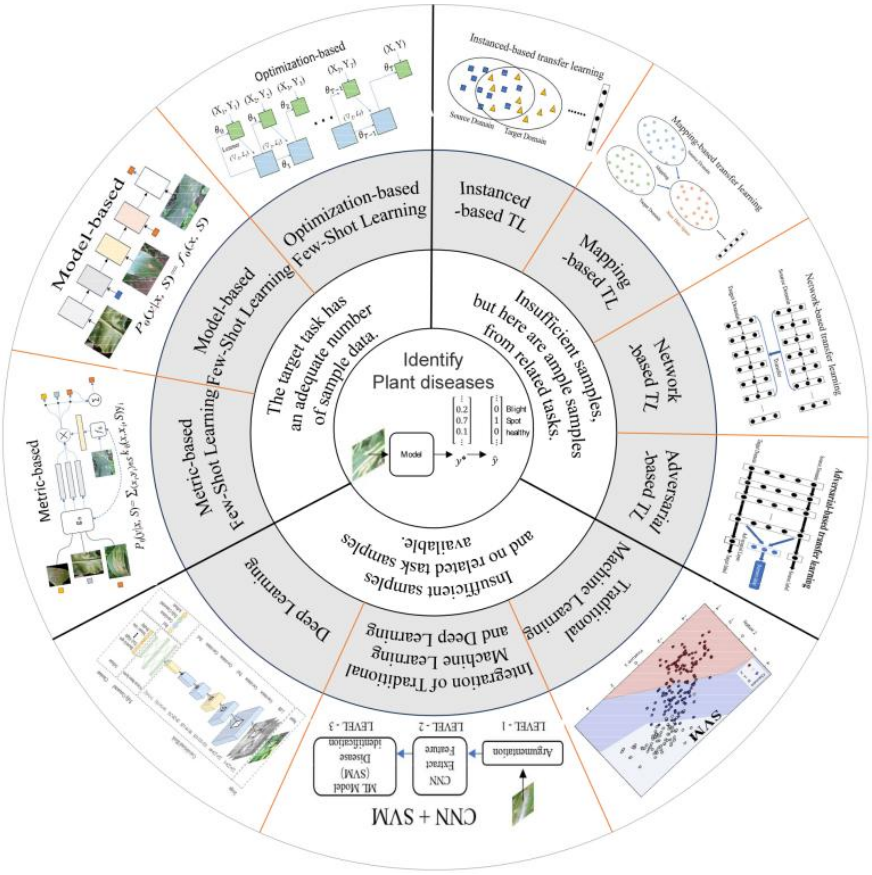


Fig. 1. Scenario-driven taxonomy of image-based plant-disease identification methods under diverse data conditions. The diagram maps three data regimes (sufficient data, limited data with related sources, and limited data without related sources) to representative technique families. (ML, DL and their fusion; transfer learning; few-shot learning).

A. Traditional Machine Learning Methods

Traditional ML methods rely on extracting features such as color, texture, and shape, making the models interpretable. Support Vector Machines (SVM) are particularly effective in handling high-dimensional data [17], while the K-Nearest Neighbors (KNN) algorithm is known for its simplicity and reliability [18]. Additionally, Naive Bayes (NB) classifiers are efficient when processing large datasets [19]. Furthermore, ensemble learning methods like Random Forests (RF) and Gradient Boosting Trees combine multiple weak learners to improve overall performance and robustness [20].

Logistic Regression (LR), NB, and Discriminant Analysis (DA) are often used to simplify complex problems by assuming feature independence, which reduces high-dimensional data to simpler forms. Appeltans *et al*. [21] trained an LR classifier to detect five classes of leek white tip disease, achieving an overall accuracy of 96.74%. However, LR tends to perform poorly in cases involving nonlinear or multi-feature problems, especially when multicollinearity among features is present [22]. DA assumes that data follows a normal distribution and that different groups have equal covariance. Mahmud *et al*. [23] used DA for real-time detection of strawberry PM disease in the field by extracting texture features with color co-occurrence matrices. This method achieved a maximum accuracy of 95.45%. However, DA may not perform well if the data does not meet the assumption of normal distribution [24]. NB works well with different types of data distributions. For instance, Prashar *et al*. [19] used NB combined with a firefly optimization method to identify 494 rice leaf images from five disease categories. However, NB may suffer from reduced accuracy when dealing with features not present in the training data or when ignoring dependencies between features.

SVM, KNN, and Decision Trees (DT) are non-parametric models that do not rely on specific data distribution assumptions or linear relationships among features. KNN works well for small datasets with low-dimensional image features [18]. Devi *et al*. [25] combined Harris corner detection, Histogram of Oriented Gradients, and KNN for peanut disease classification. This method achieved an accuracy of 97.67%. SVMs may face challenges with high-dimensional data due to the "curse of dimensionality" [26]. To address this, Javidan *et al*. [27] used Principal Component Analysis to reduce feature dimensions and applied ReliefF to select key features. RF improves model accuracy by combining the results of multiple DTs. Basavaiah *et al*. [20] compared RF and DT classifiers for plant disease detection. RF achieved a higher accuracy of 94%, while DT reached 90%. Table I summarizes traditional ML methods for plant disease identification.

TABLE I. SUMMARY OF TRADITIONAL ML METHODS FOR PLANT DISEASE IDENTIFICATION

Methods	Dataset	Type of features	Accuracy (%)	Disadvantages
RF [20]	Tomatoes, self-built	Text, color, and shape	94.00	Requires large datasets.
OWs+KNN [18]	Strawberries, self-built	Spectral	93.33	Sensitive to irrelevant features.
SLIC_SVM [17]	Tea, self-built	Shape and textural	98.50	Depends on hyperparameters.
LDA+LR [21]	White, self-built	Hyperspectral	96.74	Limited on high-dimensional.
CCM+DA [23]	Strawberries, self-built	Color and textural	95.45	Not suitable for nonlinear data.
NB+Firefly [19]	Paddy, self-built	Color and textural	98.64	Conditional independence.
SVM+FOA [26]	Tomatoes, PlantVillage	Text, color, and shape	91.10	Slow convergence.

B. Deep Learning Methods

DL enables the automatic extraction of hierarchical representations from large and complex datasets. Methods such as CNN, recurrent neural network (RNN), and Transformer effectively capture intricate patterns and features in plant imagery, offering significant advantages over traditional ML methods.

CNNs are powerful tools for extracting image features through layers of convolution, pooling, and fully connected operations. They can improve performance by increasing network depth, but deeper networks often make training more challenging. ResNet [28] addresses this issue by introducing residual connections, which facilitate gradient flow, allowing the training of much deeper models. On the PlantVillage dataset, a widely used public dataset, the PDICNet model based on ResNet-50 achieved an accuracy of 99.73% [13]. In contrast, DenseNet [29] connects each layer to every other layer in a feed-forward way, improving feature sharing and reducing redundancy. Multi-headed DenseNet was tested on the PlantVillage and achieved an accuracy of 98.17% [30].

Unlike ResNet and DenseNet, MobileNet is designed to reduce the number of parameters and computational cost. However, its accuracy may be lower for more complex datasets, as noted by Bi *et al*. [31]. They improved the MobileNet and achieved an accuracy of 73.50%. Multi-kernel depthwise separable convolutions are employed to reduce the number of model hyperparameters. Lite-MDC [32] utilizes only 2.20M parameters and achieves an accuracy of 99.78% on PlantVillage. These methods are compatible with mobile and embedded systems, where computational efficiency is critical. By focusing on diseased areas and reducing the influence of background noise, attention mechanisms contribute to improved classification accuracy [33]. The rE-GoogLeNet model adds the ECA attention mechanism into the Inception module and achieves an accuracy of 99.58% [34]. XSE-TomatoNet [35] combines SE blocks with multi-scale feature fusion to enhance the classification performance of EntricNet-B0.

For real-time applications, improved models such as YOLOv5s [36] and DM-YOLO [37] have been proposed to achieve faster inference while maintaining high detection accuracy. However, despite their efficiency in spatial object detection, these models are limited in capturing temporal dependencies across image sequences. To address this, RNNs have been employed for analyzing time-lapse images and video data, enabling the modeling of dynamic changes in plant growth and disease progression [38].

Vision Transformers (ViTs) have demonstrated strong performance in plant disease identification by modeling long-range dependencies. A recent study reported an optimized ViT architecture that attained state-of-the-art performance on PlantVillage [39]. To improve inference efficiency and reduce model complexity, recent studies have begun replacing traditional Transformer modules with Mamba-based architectures. LF-Mamba [40] improves the decoder's spatial resolution and enhances its ability to extract complex disease features, while significantly reducing the number of parameters. Table II summarizes DL-based methods for plant disease identification.

TABLE II. SUMMARY OF DL METHODS FOR PLANT DISEASE IDENTIFICATION

Methods	Dataset	Accuracy (%)	Parameters (M)	Disadvantages
Multi-headed DenseNet [30]	PlantVillage [41]	98.17	\	Higher memory requirements.
RIC-Net [42]	PlantVillage	99.55	19.10	Complex structure.
RE-GoogLeNet [34]	Rice, self-built	99.58	9.18	Difficult to optimize.
PDICNet [13]	PlantVillage	99.73	\	Difficult to optimize.
XSE-TomatoNet [35]	Tomato, PlantDoc	99.11	\	Requires high input resolution.
MobileNet [31]	Apple, self-built	73.50	\	Lower detection accuracy.
Lite-MDC [32]	PlantVillage	99.77	2.20	Unable to localize disease regions .
YOLOv5s [36]	MSMSVDD	93.10	8.30	Less effective for small objects.
DM-Yolo [37]	Tomato, self-built	92.50	50.07	Lower training stability.
G-RecConNN [38]	Banana, self-built	93.60	\	Vanishing gradient issues.
Optimized ViT [39]	PlantVillage	99.77	13.02	More computational resources.
LeafMamba [40]	FGVC8	92.50	0.99	Limited capability in local feature.

C. Hybrid Methods

Hybrid ensemble methods leverage the complementary strengths of different architectures to improve image feature extraction, mitigate overfitting caused by feature bias, and enhance generalization [53]. To balance performance and efficiency, recent studies have explored combining CNN-based feature extractors with traditional classifiers, integrating CNNs' spatial feature learning with the interpretability of classical methods [43],[44]. Meanwhile, many hybrid methods reduce model complexity by replacing parts of complex architectures with lightweight components, while maintaining or even enhancing predictive performance [45]. For example, SSD_Mobile-NetV2 [46] uses only 50.9% of the parameters compared to SSD_VGG, yet achieves improvements in precision, recall, F1-score, and mean Average Precision (mAP) by 4.37%, 3.3%, 3.8%, and 8.79%, respectively. Furthermore, CNNs are often employed to replace or supplement parts of Transformer architectures, providing more efficient local feature modeling and faster inference in real-time or resource-constrained plant disease detection scenarios. ConvViT [47], which combines CNNs with Transformer, outperforms models such as ResNet, ViT, and ResMLP by improving accuracy by 4.53% while reducing the number of parameters and FLOPs by more than 10%.

SoyaTrans [48] utilizes CNNs to extract local features and a Transformer module to capture global dependencies. With only 5.2 million parameters, it achieves an accuracy of 98.00% on the PlantVillage dataset. This result demonstrates that integrating multiple learning paradigms can outperform models based on a single architecture. Table III summarizes the latest performance of hybrid methods in plant disease identification tasks.

TABLE III. SUMMARY OF RECENT PERFORMANCE OF HYBRID METHODS IN PLANT DISEASE IDENTIFICATION

Ensemble Methods	Dataset	Accuracy (%)	Parameters (M)
CNN+Transformer [47]	PlantVillage	99.84	59.00
VGG19+InceptionV3 [49]	PlantVillage	98.66	\
VGG16 + RF [43]	PlantVillage	94.00	0.26
KNN+ Transformer [44]	PlantVillage	99.93	\
SoyaTrans [48]	PlantVillage	98.00	5.20
VGG-ICNN [50]	PlantVillage	99.16	23.20
CNN + RNN [51]	AI maize dataset	97.85	0.4
Ensemble models [52]	New PlantVillage	99.89	\

D. Discussion and Summary

Traditional ML, DL, and hybrid methods provide effective solutions for plant disease classification. The choice of method depends on the specific challenges of the task, such as data availability, feature complexity, and computational resources. Traditional ML methods are lightweight and interpretable due to their reliance on handcrafted features. However, their effectiveness is limited by feature quality and poor robustness under varying field conditions. DL methods automatically extract features from raw data. They are powerful at capturing complex disease patterns and usually achieve higher accuracy than ML when sufficient labeled data is available. However, they are data- and resource-intensive, less interpretable, and often struggle with robustness under real-world field conditions. Hybrid ML-DL methods combine the strengths of both paradigms. DL models are used for robust feature extraction, while traditional ML algorithms provide lightweight and interpretable decision-making. This balance reduces reliance on manual features and lowers computational cost compared with full DL pipelines. However, their effectiveness is constrained by the need for precise feature-model integration and limited robustness to previously unseen crops and disease types [54].

III. TRANSFER LEARNING FOR PLANT DISEASE IDENTIFICATION

TL allows models trained on one plant species to be applied to related species. It helps address limited data and domain variability, especially under class imbalance [55]. However, a comprehensive review specifically focused on “agricultural TL” remains lacking [56]. This section reviews three major TL methods, namely instance-based, mapping-based, and network-based.

A. Mapping-Based Transfer Learning

Mapping-based TL reduces the distribution gap between source and target domains by learning a shared feature space or an explicit mapping function. MViT [57] employs an adversarial framework augmented with the Wasserstein distance to learn domain-invariant representations, projecting source and target samples into a common latent space with closely matched distributions. Maximum Mean Discrepancy [58] offers a theoretically grounded measure of domain shift by aligning the mean embeddings of source and target features in a reproducing-kernel Hilbert space. CLA [59] first performs self-supervised contrastive pre-training, then applies a domain-adaptation layer that explicitly aligns the resulting feature distributions. CORrelation Alignment [60] efficiently matches the second-order statistics of source and target features, delivering low computational overhead and good scalability. Attention-based alignment [61] further narrows the domain gap by highlighting discriminative regions while suppressing irrelevant background noise.

For cross-species plant disease identification, adversarial domain adaptation is used to learn domain-invariant representations. A representative method is DANN [62], which performs well under pronounced domain shifts, including highly non-linear settings and large cross-domain datasets. Other strategies reduce domain discrepancy by explicitly constraining the diagnostic model. FMDA [63] minimises the “distance” between source and target, thereby lowering domain divergence. Using unlabelled target data without explicit feature alignment can exacerbate domain mismatch. Unsupervised domain adaptation methods [64] learn shared feature spaces from abundant unlabelled target images, aligning source and target domains through non-adversarial means. Zero-shot TL [65] builds a discriminative, transferable embedding that bridges source and target domains even when their class sets do not overlap.

B. Network-Based Transfer Learning

Network-based TL focuses on modifying the architecture itself so that it can more effectively adapt to new tasks or domains. These methods exploit the learned by a pre-trained backbone, and enhance identification performance through optimizing the architecture [66]. In addition to direct architectural adjustments, several studies have reported further gains in overall accuracy, generalization, and overfitting resistance by employing multi-model fusion or model ensembling [67]. INC-VGGN [68] integrates a pre-trained VGGNet with an Inception module, combining VGG's ability to capture deep hierarchical representations with Inception's multi-scale feature extraction, thereby improving the network's capacity to model fine-grained details. SLViT [69] merges a Transformer encoder with a CNN, striking a balance between recognition performance and computational cost. The incorporation of attention mechanisms has opened an additional optimization avenue. For example, Sudhesh *et al.* [70] introduced an attention module into a DenseNet121 with an RF framework, reinforcing the network's focus on disease-relevant regions.

Unlike single-stage TL, two-stage TL introduces an intermediate adaptation phase that further tailors a network to the target task. Es-MbNet [71], for example, re-trains on the target data, leading to a marked performance gain, after an initial weight initialization stage. A similar method keeps the first ten layers of a pre-trained VGG-16 network fixed, appends two inception blocks, and then fine-tunes the modified architecture with target data, reaching 99.23% accuracy [72]. In addition, knowledge distillation is emerging as an effective transfer-learning technique. By distilling the feature representations of a pre-trained Vision Transformer into a lightweight student model, this method significantly reduces model complexity while maintaining competitive performance [73].

C. Instance-Based Transfer Learning

Instance-based TL fine-tunes models that have been pre-trained on large-scale datasets such as ImageNet and reuses them as feature extractors, thereby enabling efficient knowledge transfer to a target task [74]. For example, a pre-trained VGG16 optimized with stochastic gradient descent with momentum (SGDm) achieved 98.33% accuracy in distinguishing healthy fruit from fruit infected by the Mediterranean fruit fly [75]. A pre-trained MobileNetV2 reached 99.30 % accuracy in detecting tomato leaf diseases on the PlantVillage [76]. A fine-tuned VGG-16 attained 95.00% classification accuracy across 11 tomato disease classes [77]. Even on very small datasets, the approach remains effective. Yu *et al*. [78] fine-tuned a pre-trained ResNet-18 on a dataset of only 120 images and achieved 99.53 % accuracy, while Liu *et al.* [79] reported 99.45% accuracy for apple-disease recognition using a pre-trained Inception-V3 model.

TABLE IV. SUMMARY OF TL METHODS FOR PLANT DISEASE IDENTIFICATION (%)

Model Structure Basis	Source Domain	Target Domain	Accuracy	Recall	F1	Precision
MViT [57]	PlantVillage	PlantDoc [80]	70.00	69.00	67.00	68.00
CPAM-DSAN [61]	Dataset300	Rice, self-built	95.25	95.25	95.34	95.38
Transformer [73]	ImageNet	Rice, self-built	92.00	91.00	89.00	88.00
ResNet18 [78]	ImageNet	Soybean, self-built	99.53	99.55	99.54	99.53
Inception-V3 [79]	ImageNet	PlantVillage	99.45	99.10	99.00	98.84
MobileNetV2 [76]	ImageNet	PlantVillage	99.30	\	\	\
VGG19 [67]	ImageNet	rice, self-built	93.00	89.90	90.50	92.40
XceptionNet [70]	\	Rice, self-built	93.87	85.00	85.00	85.00
VGG16 [77]	ImageNet	tomato, public	95	93.55	94.00	94.55
ResNet50 [81]	2018 AI challenge	2018 AI challenge	97.30	\	\	\
VGG16 [75]	ImageNet	Citrus, self-built	98.33	99.33	99.34	99.35
VGG19 [68]	ImageNet	PlantVillage	91.83	\	\	\
VGG16 [72]	ImageNet	Tomato，self-built	99.23	99.29	99.20	99.12
Es-MbNet [71]	ImageNet	PlantVillage	99.61	98.08	98.08	\
VGG16 [66]	ImageNet	Rice and wheat, self-built	97.99	\	\	\
DANet [82]	CPD P1, self-built	CPD P2, self-built	96.75	\	\	\
STA-GAN [83]	ImageNet	Cucumber, self-built	98.97	\	\	\
AlexNet [84]	ImageNet	PlantVillage	96.63	91.00	91.00	92.00
DTL-SE-ResNet50 [85]	AI Challenger 2018	Vegetable, self-built	97.24	92.58	94.85	\

D. Discussion and Summary

The choice of TL paradigm ultimately reflects trade-offs between resource availability, domain similarity, and the complexity of the target task. Instance-based TL fine-tunes pre-trained models with minimal changes, making it lightweight and suitable for resource-limited tasks when domains are closely aligned. Its drawback is strong dependence on domain similarity and risk of negative transfer under large gaps. Mapping-based TL aligns source and target features via

statistical measures, adversarial training, or attention. It is effective when related source data are abundant but target data are scarce, though it often adds complexity, unstable training, and low interpretability under large domain shifts. Network-based TL adapts architectures for deeper transfer, yielding strong performance in complex or fine-grained tasks with large domain differences, but at the expense of high computational and data demands. Table IV summarizes various transfer learning techniques and their effectiveness in plant disease identification, highlighting the source and target domains, accuracy, recall, Precision, F1 scores, and model architectures.

IV. Few-Shot Learning for Plant Disease Identification

ML, DL, and TL methods typically require a large amount of labeled data for model training, but in practical applications, obtaining such data can be very challenging or expensive. FSL Methods aim to address the issue of learning and generalizing in the absence of labeled samples.

A. Model-Based Few-Shot Learning

The limited availability of labeled samples often results in significant class imbalance, which poses a major challenge for plant disease identification models. To address this issue, researchers have explored synthetic data generation methods to enhance prediction performance, such as Variational Autoencoders [86],[87], GANs [88], [89], and Diffusion Models [90], [91] are widely used to generate new samples. For example, Singh *et al*. [92] used a Conditional GAN to synthesize maize leaf images for data augmentation, which enhanced classification performance compared with using real images alone. Texture Reconstruction Loss Cycle GAN [93] to generate realistic citrus Huanglongbing leaf data. This method improved the average accuracy of the classification network by 2.76%.

To improve model robustness, CNN_BIT [94] integrates a CNN with BiT, combining their feature extraction capabilities to address a few-shot learning task involving only 20 images per class. Some methods combined generative models with discriminative models. Generating more diversified plant disease images through training to improve the identification model's performance [95]. Sharma *et al*. [96] developed a conditional deep convolutional GAN (ClGan) with an improved loss and paired it with a compact classifier (ClGanNet) for maize leaves. It achieved 99.04% test accuracy and strong parameter efficiency. However, these methods struggle with limited adaptability and poor generalization under task distribution shifts.

B. Metric-Based Few-Shot Learning

Metric-based methods aim to learn a similarity metric to measure the similarity between samples. Prototypical Networks perform class prototype computation and distance-based classification [97]. Common distance metrics such as Euclidean distance and cosine similarity are widely used to compare feature representations between support and query samples in few-shot learning. A common strategy is to train the CNN-based feature extractor with supervised contrastive objectives, which pull together embeddings from the same class while pushing apart those from different classes under episodic sampling [98]. Some methods further enhance this approach by integrating triplet loss with efficient class-boundary shallow learners, improving classification performance in low-data scenarios [99]. Alternatively, Circle Loss [100] was introduced to measure sample similarity by optimizing the relative distances among positive, anchor, and negative samples in the embedding space, leading to better feature discrimination. MAFDE-DN4 [101] enhances inter-class separability in the embedding space by introducing additional discriminative distances for each class. However, these methods rely heavily on well-structured embeddings and are limited by intra-class variability and sensitivity to distance metrics.

C. Optimization-Based Few-Shot Learning

Optimization-based methods find the optimal decision boundary to distinguish different classes on a limited sample of instances. Meta-learning (learn-to-learn) methods [102] fall under this category, training a meta-learner to quickly adapt to new tasks using only a few training samples. Through transfer optimization, parameters or features of pre-trained models can be optimized for better generalization ability on few-shot tasks [103]. PMF+FA [104] integrates pre-training, meta-learning, and fine-tuning (PMF) with a novel Feature Attention module. This module highlights discriminative regions within images while suppressing the influence of complex backgrounds and irrelevant objects. However, these methods are hindered by initialization sensitivity, high computational cost from inner-loop updates, and slow convergence in complex, non-convex spaces.

D. Discussion and Summary

FSL methods become increasingly useful for annotated data in plant disease identification. Their choice depends on the balance between data scarcity, task diversity, and available resources. Model-based FSL methods design architectures that rapidly adapt to new tasks by leveraging shared knowledge across episodes. They are effective for learning transferable representations and provide strong performance when tasks are diverse, but they often involve complex designs and high training costs. Metric-based FSL methods emphasize similarity between support and query samples using distance functions or embedding spaces. They are lightweight, intuitive, and well-suited for cases where disease features are visually consistent, but their performance degrades when inter-class similarities are high or intra-class variations are large. Optimization-based FSL methods aim to improve task adaptation through specialized training strategies, such as meta-optimization or parameter initialization schemes. They can achieve superior accuracy under scarce data, but are computationally demanding and sensitive to hyperparameter tuning. Table V summarizes the recent performances of FSL methods applied to the identification of plant diseases.

TABLE V. RECENT PERFORMANCES OF FSL IN PLANT DISEASES IDENTIFICATION

Methods	Backbone	Datasets	N-way	K = 1	K = 5	K = 10
Attn-CutMix [102]	Transformer	Cotton, self-built	3	72.90	88.00	\
Inception+SVM [99]	Inception V3	PlantVillage	6	55.30	72.60	77.00
ViT ProtoNet+FA [104]	ViT	PlantDoc, Public	5	\	86.79	\
Swin-B [105]	Transformer	Barley, self-built	5	38.63	66.46	78.15
PruneFSL [106]	Resnet 12	Plantvillage	5	77.97	90.70	\
ResNet18+ProtoNet [97]	ResNet18	PlantVillage	10	72.76	92.12	93.32
MAFDE-DN4 [101]	ResNet 12	PlantVillage	5	57.50	81.41	\
MixNet-XL [100]	MixNet-XL	PlantVillage	6	75.41	91.03	93.97
SC-FSL [98]	Resnet 18	PlantVillage	5	78.55	92.90	\
CNN + SVM [103]	GoogleNet	PlantVillage	38	\	75.20	88.40
STV2F6+PDFC [107]	STV2F6	PlantVillage	5	91.81	95.32	95.50

V. FUTURE DIRECTIONS

Advancements in plant disease recognition technology are expected to continue driving significant progress in agriculture and contributing substantially to crop protection and management. Based on recent trends and developments, several promising directions and opportunities emerge.

1) Future research will increasingly harness multi-modal data by integrating visual imagery, audio signals, spectral information, textual inputs, and environmental parameters. This convergence of diverse data modalities will enable more comprehensive feature extraction, thereby enhancing the accuracy of pest and disease detection and classification. Advanced language models such as Claude, GPT-4, Gemini, and DeepSeek can play a pivotal role by combining visual inputs with descriptive textual data to improve early identification and nuanced assessment of plant health [108]. Additionally, integrating visible-spectrum imagery with infrared sensor data can significantly enhance the precision and timeliness of plant stress detection [109].

2) Few-Shot Domain Adaptation (FSDA): FSDA has demonstrated significant potential for addressing the substantial distribution differences and limited labeled data commonly encountered in real-world scenarios. Future research will focus on improving cross-task generalization, enabling models trained on source domains to effectively transfer to target domains with minimal or no labeled data, thereby alleviating performance bottlenecks caused by data scarcity [110]. Additionally, prototype aggregation methods further enhance robustness in few-shot and cross-domain scenarios by aggregating prototypes from multiple samples or domains, thus capturing finer intra-class structures and subtle differences [111].

3) Diffusion model-based sample generation: Utilizing diffusion models for generating interpretable synthetic samples represents an innovative direction in future research [112]. By generating interpretable synthetic samples, diffusion models provide intuitive visualizations that reveal the critical features learned by models, highlight potential biases, and uncover blind spots in training data. Additionally, in few-shot domain adaptation tasks, diffusion models guided by limited target domain samples can generate high-quality pseudo-samples, effectively expanding the target domain distribution and alleviating label scarcity [113]. Future research will explore designing more semantically consistent and finely controlled conditional diffusion mechanisms, alongside advancements in efficient sampling and low-resource deployment, ultimately driving practical applications in high-value fields such as medical imaging and agricultural remote sensing.

4) Edge computing and mobile applications: With the proliferation of mobile technologies and edge computing infrastructure, plant pest and disease identification tools are expected to become increasingly accessible and widely adopted. Deploying sophisticated recognition models directly onto mobile and edge devices will facilitate real-time disease diagnosis, significantly reducing data transmission costs, minimizing latency, and improving overall responsiveness and usability in agricultural field environments [114].

5) Model interpretability: In agricultural applications such as plant pest and disease recognition, model interpretability is becoming a critical factor for enhancing system reliability and practical usability. Future research will focus on embedding interpretability mechanisms into lightweight models, enabling techniques such as Grad-CAM to function efficiently on mobile and edge devices to support real-time, in-field diagnostics [115]. At the same time, concept-based interpretability methods are expected to advance by identifying high-level semantic features, such as "leaf yellowing" or "marginal decay," which align closely with agricultural domain knowledge and improve the clarity and usefulness of model outputs [116]. Additionally, multimodal interpretability is likely to become a major research direction by integrating image data with sensor information and environmental context.

VI. CONCLUSION

Image-based methods for plant pest and disease classification hold significant potential in improving detection accuracy, which is crucial for early disease prediction and timely intervention. This review explores various approaches

in plant disease recognition through image analysis, covering traditional ML, DL, and hybrid methods. We explore their achievements and inherent challenges, such as data imbalance, complex training requirements, and domain adaptability. TL and FSL emerge as effective solutions to address the limited availability of annotated data, each offering distinct advantages and facing unique limitations. TL leverages pre-trained models from related domains to improve plant disease detection, with Mapping-based and Network-based TL showing high accuracy in tasks like tomato and soybean leaf disease classification. However, Instance-based TL is less effective due to issues with data distribution and domain mismatches. FSL trains models with minimal labeled data. Model-based FSL methods, such as those using GANs, generate synthetic data to enhance model generalization. Metric-based FSL, including Prototypical and Matching Networks, excels in classification tasks by learning similarity measures between labeled and unlabeled data. Optimization-based FSL enables rapid adaptation to new tasks but faces challenges such as overfitting and hyperparameter sensitivity.

REFERENCES

[1] Z. Yang et al., "FATDNet: A fusion adversarial network for tomato leaf disease segmentation under complex backgrounds," Computers and Electronics in Agriculture, vol. 234, p. 110270, Jul. 2025.

[2] G. Singh, K. Guleria, and S. Sharma, "Leveraging Transfer Learning-based Fine-tuned ResNet50 Model for Maize Leaf Disease Classification," in 2024 5th International Conference for Emerging Technology (INCET), Belgaum, India: IEEE, May 2024, pp. 1–6.

[3] W. B. Demilie, "Plant disease detection and classification techniques: a comparative study of the performances," J Big Data, vol. 11, no. 1, p. 5, Jan. 2024.

[4] A. Guerrero-Ibañez and A. Reyes-Muñoz, "Monitoring tomato leaf disease through convolutional neural networks," Electronics, vol. 12, no. 1, p. 229, 2023.

[5] S. Dananjayan, Y. Tang, J. Zhuang, C. Hou, and S. Luo, "Assessment of state-of-the-art deep learning based citrus disease detection techniques using annotated optical leaf images," Comput Electron Agr, vol. 193, p. 106658, Feb. 2022.

[6] J. Qi et al., "An improved YOLOv5 model based on visual attention mechanism: Application to recognition of tomato virus disease," Comput Electron Agr, vol. 194, p. 106780, Mar. 2022.

[7] M. Azadbakht, D. Ashourloo, H. Aghighi, S. Radiom, and A. Alimohammadi, "Wheat leaf rust detection at canopy scale under different LAI levels using machine learning techniques," Computers and Electronics in Agriculture, vol. 156, pp. 119–128, Jan. 2019.

[8] R. Shi, D. Zhai, X. Liu, J. Jiang, and W. Gao, "Rectified Meta-Learning from Noisy Labels for Robust Image-based Plant Disease Diagnosis," Mar. 18, 2020, arXiv: arXiv:2003.07603.

[9] S. Verma, P. Kumar, and J. P. Singh, "MLNAS: Meta-learning based neural architecture search for automated generation of deep neural networks for plant disease detection tasks," Network: Computation in Neural Systems, pp. 1–24, Jul. 2024.

[10] Z. Tang et al., "Affordable High Throughput Field Detection of Wheat Stripe Rust Using Deep Learning with Semi-Automated Image Labeling," Comput Electron Agr, vol. 207, p. 107709, Apr. 2023.

[11] X. Jin et al., "TranNas-NirCR: A method for improving the diagnosis of asymptomatic wheat scab with transfer learning and neural architecture search," Computers and Electronics in Agriculture, vol. 213, p. 108271, Oct. 2023.

[12] Z. H. Kok, A. R. Mohamed Shariff, M. S. M. Alfatni, and S. Khairunniza-Bejo, "Support Vector Machine in Precision Agriculture: A review," Comput Electron Agr, vol. 191, p. 106546, Dec. 2021.

[13] S. R. Reddy, G. S. Varma, and R. L. Davuluri, "Resnet-based modified red deer optimization with DLCNN classifier for plant disease identification and classification," Computers and Electrical Engineering, vol. 105, p. 108492, 2023.

[14] S. H. Lee, H. Goëau, P. Bonnet, and A. Joly, "New perspectives on plant disease characterization based on deep learning," Computers and Electronics in Agriculture, vol. 170, p. 105220, Mar. 2020.

[15] Y. Lu, D. Chen, E. Olaniyi, and Y. Huang, "Generative adversarial networks (GANs) for image augmentation in agriculture: A systematic review," Comput Electron Agr, vol. 200, p. 107208, Sep. 2022.

[16] W. Ding, M. Abdel-Basset, I. Alrashdi, and H. Hawash, "Next generation of computer vision for plant disease monitoring in precision agriculture: A contemporary survey, taxonomy, experiments, and future direction," Information Sciences, vol. 665, p. 120338, Apr. 2024.

[17] Y. Sun, Z. Jiang, L. Zhang, W. Dong, and Y. Rao, "SLIC_SVM based leaf diseases saliency map extraction of tea plant," Comput Electron Agr, vol. 157, pp. 102–109, Feb. 2019.

[18] G. Wu et al., "Early identification of strawberry leaves disease utilizing hyperspectral imaging combing with spectral features, multiple vegetation indices and textural features," Comput Electron Agr, vol. 204, p. 107553, Jan. 2023.

[19] V. Prashar, D. Sood, and S. Kumar, "An Efficient Recognition and Classification System for Paddy Leaf Disease Using Naïve Bayes with Optimization Algorithm," in Proceedings of International Conference on Recent Innovations in Computing, vol. 1011, Y. Singh, C. Verma, I. Zoltán, J. K. Chhabra, and P. K. Singh, Eds., in Lecture Notes in Electrical Engineering, vol. 1011. , Singapore: Springer Nature Singapore, 2023, pp. 149–167.

[20] J. Basavaiah and A. Arlene Anthony, "Tomato Leaf Disease Classification using Multiple Feature Extraction Techniques," Wireless Pers Commun, vol. 115, no. 1, pp. 633–651, Nov. 2020.

[21] S. Appeltans, J. G. Pieters, and A. M. Mouazen, "Detection of leek white tip disease under field conditions using hyperspectral proximal sensing and supervised machine learning," Comput Electron Agr, vol. 190, p. 106453, Nov. 2021.

[22] M. B. Almoujahed et al., "Detection of fusarium head blight in wheat under field conditions using a hyperspectral camera and machine learning," Comput Electron Agr, vol. 203, p. 107456, Dec. 2022.

[23] Md. Sultan Mahmud, Q. U. Zaman, T. J. Esau, G. W. Price, and B. Prithiviraj, "Development of an artificial cloud lighting condition system using machine vision for strawberry powdery mildew disease detection," Computers and Electronics in Agriculture, vol. 158, pp. 219–225, Mar. 2019.

[24] G. Xuan, Q. Li, Y. Shao, and Y. Shi, "Early diagnosis and pathogenesis monitoring of wheat powdery mildew caused by blumeria graminis using hyperspectral imaging," Computers and Electronics in Agriculture, vol. 197, p. 106921, Jun. 2022.

[25] K. Suganya Devi, P. Srinivasan, and S. Bandhopadhyay, "H2K – A robust and optimum approach for detection and classification of groundnut leaf diseases," Computers and Electronics in Agriculture, vol. 178, p. 105749, Nov. 2020.

[26] E. Gangadevi, R. S. Rani, R. K. Dhanaraj, and A. Nayyar, "Spot-out fruit fly algorithm with simulated annealing optimized SVM for detecting tomato plant diseases," Neural Comput & Applic, vol. 36, no. 8, pp. 4349–4375, Mar. 2024.

[27] S. M. Javidan, A. Banakar, K. A. Vakilian, and Y. Ampatzidis, "Diagnosis of grape leaf diseases using automatic K-means clustering and machine learning," Smart Agricultural Technology, vol. 3, p. 100081, Feb. 2023.

[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA: IEEE, Jun. 2016, pp. 770–778.

[29] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," Jan. 28, 2018, arXiv: arXiv:1608.06993.

[30] Y. Kaya and E. Gürsoy, "A novel multi-head CNN design to identify plant diseases using the fusion of RGB images," Ecological Informatics, vol. 75, p. 101998, Jul. 2023.

[31] C. Bi, J. Wang, Y. Duan, B. Fu, J.-R. Kang, and Y. Shi, "MobileNet Based Apple Leaf Diseases Identification," Mobile Netw Appl, vol. 27, no. 1, pp. 172–180, Feb. 2022.

[32] S. Bhagat et al., "Advancing real-time plant disease detection: A lightweight deep learning approach and novel dataset for pigeon pea crop," Smart Agricultural Technology, vol. 7, p. 100408, Mar. 2024.

[33] F. Tang, R. R. Porle, H. Tung Yew, and F. Wong, "Identification of Maize Diseases Based on Dynamic Convolution and Tri-Attention Mechanism," IEEE Access, vol. 13, pp. 6834–6844, 2025.

[34] L. Yang et al., "GoogLeNet based on residual network and attention mechanism identification of rice leaf diseases," Comput Electron Agr, vol. 204, p. 107543, Jan. 2023.

[35] M. Assaduzzaman, P. Bishshash, Md. A. S. Nirob, A. A. Marouf, J. G. Rokne, and R. Alhajj, "XSE-TomatoNet: An explainable AI based tomato leaf disease classification method using EfficientNetB0 with squeeze-and-excitation blocks and multi-scale feature fusion," MethodsX, vol. 14, p. 103159, Jun. 2025.

[36] J. Li, Y. Qiao, S. Liu, J. Zhang, Z. Yang, and M. Wang, "An improved YOLOv5-based vegetable disease detection method," Computers and Electronics in Agriculture, vol. 202, p. 107345, Nov. 2022.

[37] A. Abulizi, J. Ye, H. Abudukelimu, and W. Guo, "DM-YOLO: improved YOLOv9 model for tomato leaf disease detection," Front. Plant Sci., vol. 15, p. 1473928, Feb. 2025.

[38] M. Nandhini, K. U. Kala, M. Thangadarshini, and S. Madhusudhana Verma, "Deep Learning model of sequential image classifier for crop disease detection in plantain tree cultivation," Comput Electron Agr, vol. 197, p. 106915, Jun. 2022.

[39] A. Ouamane et al., "Optimized Vision Transformers for Superior Plant Disease Detection," IEEE Access, vol. 13, pp. 48552–48570, 2025.

[40] X. Wang, S. H. Tang, M. K. A. B. M. Ariffin, M. I. S. B. Ismail, and R. Zhao, "LeafMamba: A novel IoT-integrated network for accurate and efficient plant leaf disease detection," Alexandria Engineering Journal, vol. 123, pp. 415–424, Jun. 2025.

[41] D. Hughes and M. Salathé, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," arXiv preprint arXiv:1511.08060, 2015.

[42] Y. Zhao, C. Sun, X. Xu, and J. Chen, "RIC-Net: A plant disease classification model based on the fusion of Inception and residual structure and embedded attention mechanism," Comput Electron Agr, vol. 193, p. 106644, Feb. 2022.

[43] Y. Li, J. Nie, and X. Chao, "Do we really need deep CNN for plant diseases identification?," Computers and Electronics in Agriculture, vol. 178, p. 105803, Nov. 2020.

[44] H. Liu, Y. Zhan, H. Xia, Q. Mao, and Y. Tan, "Self-supervised transformer-based pre-training method using latent semantic masking auto-encoder for pest and disease classification," Comput Electron Agr, vol. 203, p. 107448, Dec. 2022.

[45] P. Dhoundiyal, V. Sharma, S. Vats, and P. Rawat, "A Progressive Hierarchical Model for Plant Disease Diagnosis," SN COMPUT. SCI., vol. 6, no. 2, p. 102, Jan. 2025.

[46] W. Guo, S. Feng, Q. Feng, X. Li, and X. Gao, "Cotton leaf disease detection method based on improved SSD," International Journal of Agricultural and Biological Engineering, vol. 17, no. 2, pp. 211–220, 2024.

[47] X. Li, X. Chen, J. Yang, and S. Li, "Transformer helps identify kiwifruit diseases in complex natural environments," Comput Electron Agr, vol. 200, p. 107258, Sep. 2022.

[48] V. Sharma, A. K. Tripathi, H. Mittal, and L. Nkenyereye, "SoyaTrans: A novel transformer model for fine-grained visual classification of soybean leaf disease diagnosis," Expert Systems with Applications, vol. 260, p. 125385, Jan. 2025.

[49] F. Arshad et al., "PLDPNet: End-to-end hybrid deep learning framework for potato leaf disease prediction," Alex Eng J, vol. 78, pp. 406–418, Sep. 2023.

[50] P. S. Thakur, T. Sheorey, and A. Ojha, "VGG-ICNN: A Lightweight CNN model for crop disease identification," Multimed Tools Appl, vol. 82, no. 1, pp. 497–520, Jan. 2023.

[51] C. Ashwini and V. Sellam, "An optimal model for identification and classification of corn leaf disease using hybrid 3D-CNN and LSTM," Biomedical Signal Processing and Control, vol. 92, p. 106089, Jun. 2024.

[52] A. H. Ali, A. Youssef, M. Abdelal, and M. A. Raja, "An ensemble of deep learning architectures for accurate plant disease classification," Ecological Informatics, vol. 81, p. 102618, Jul. 2024.

[53] W. Shafik, A. Tufail, C. Liyanage De Silva, and R. A. Awg Haji Mohd Apong, "A novel hybrid inception-xception convolutional neural network for efficient plant disease classification and detection," Sci Rep, vol. 15, no. 1, p. 3936, Jan. 2025.

[54] I. Bouacida, B. Farou, L. Djakhdjakha, H. Seridi, and M. Kurulay, "Innovative deep learning approach for cross-crop plant disease detection: A generalized method for identifying unhealthy leaves," Information Processing in Agriculture, vol. 12, no. 1, pp. 54–67, Mar. 2025.

[55] I. Khan, S. S. Sohail, D. Ø. Madsen, and B. K. Khare, "Deep transfer learning for fine-grained maize leaf disease classification," Journal of Agriculture and Food Research, vol. 16, p. 101148, Jun. 2024.

[56] M. I. Hossen, M. Awrangjeb, S. Pan, and A. A. Mamun, "Transfer learning in agriculture: a review," Artif Intell Rev, vol. 58, no. 4, p. 97, Jan. 2025.

[57] M. H. Tunio et al., "Advancing plant disease classification: A robust and generalized approach with transformer-fused convolution and Wasserstein domain adaptation," Computers and Electronics in Agriculture, vol. 227, p. 109574, Dec. 2024.

[58] N. Bharathi Raja and P. Selvi Rajendran, "An efficient banana plant leaf disease classification using optimal ensemble deep transfer network," Journal of Experimental & Theoretical Artificial Intelligence, vol. 37, no. 4, pp. 585–608, May 2025.

[59] R. Zhao, Y. Zhu, and Y. Li, "CLA: A self-supervised contrastive learning method for leaf disease identification with domain adaptation," Computers and Electronics in Agriculture, vol. 211, p. 107967, Aug. 2023.

[60] L. Feng, B. Wu, Y. He, and C. Zhang, "Hyperspectral Imaging Combined With Deep Transfer Learning for Rice Disease Detection," Front. Plant Sci., vol. 12, p. 693521, Sep. 2021.

[61] L. Chen, J. Zou, Y. Yuan, and H. He, "Improved domain adaptive rice disease image recognition based on a novel attention mechanism," Computers and Electronics in Agriculture, vol. 208, p. 107806, May 2023, doi: 10.1016/j.compag.2023.107806.

[62] M. Hu, S. Long, C. Wang, and Z. Wang, "Leaf disease detection using deep Convolutional Neural Networks," J. Phys.: Conf. Ser., vol. 2711, no. 1, p. 012020, Feb. 2024.

[63] S. Kudo, S. Kagiwada, and H. Iyatomi, "Few-shot Metric Domain Adaptation: Practical Learning Strategies for an Automated Plant Disease Diagnosis," Dec. 25, 2024, arXiv: arXiv:2412.18859.

[64] X. Wu, X. Fan, P. Luo, S. D. Choudhury, T. Tjahjadi, and C. Hu, "From Laboratory to Field: Unsupervised Domain Adaptation for Plant Disease Recognition in the Wild," Plant Phenomics, vol. 5, p. 0038, 2023.

[65] R. Satya Rajendra Singh and R. K. Sanodiya, "Zero-Shot Transfer Learning Framework for Plant Leaf Disease Classification," IEEE Access, vol. 11, pp. 143861–143880, 2023.

[66] Z. Jiang, Z. Dong, W. Jiang, and Y. Yang, "Recognition of rice leaf diseases and wheat leaf diseases based on multi-task deep transfer learning," Comput Electron Agr, vol. 186, p. 106184, Jul. 2021.

[67] R. Dogra, S. Rani, A. Singh, M. A. Albahar, A. E. Barrera, and A. Alkhayyat, "Deep learning model for detection of brown spot rice leaf disease with smart agriculture," Computers and Electrical Engineering, vol. 109, p. 108659, Jul. 2023.

[68] J. Chen, J. Chen, D. Zhang, Y. Sun, and Y. A. Nanehkaran, "Using deep transfer learning for image-based plant disease identification," Computers and Electronics in Agriculture, vol. 173, p. 105393, Jun. 2020.

[69] X. Li, X. Li, S. Zhang, G. Zhang, M. Zhang, and H. Shang, "SLViT: Shuffle-convolution-based lightweight Vision transformer for effective diagnosis of sugarcane leaf diseases," Journal of King Saud University - Computer and Information Sciences, vol. 35, no. 6, p. 101401, Jun. 2023.

[70] S. K.M., S. V., S. K. P., and S. O.K., "AI based rice leaf disease identification enhanced by Dynamic Mode Decomposition," Engineering Applications of Artificial Intelligence, vol. 120, p. 105836, Apr. 2023.

[71] J. Chen, A. Zeb, Y. A. Nanehkaran, and D. Zhang, "Stacking ensemble model of deep learning for plant disease recognition," J Ambient Intell Human Comput, vol. 14, no. 9, pp. 12359–12372, Sep. 2023.

[72] T. Sanida, A. Sideris, M. V. Sanida, and M. Dasygenis, "Tomato leaf disease identification via two–stage transfer learning approach," Smart Agricultural Technology, vol. 5, p. 100275, Oct. 2023.

[73] C. Zhou, Y. Zhong, S. Zhou, J. Song, and W. Xiang, "Rice leaf disease identification by residual-distilled transformer," Engineering Applications of Artificial Intelligence, vol. 121, p. 106020, May 2023.

[74] H. Catal Reis and V. Turk, "Integrated deep learning and ensemble learning model for deep feature-based wheat disease detection," Microchemical Journal, vol. 197, p. 109790, Feb. 2024.

[75] R. Hadipour-Rokni, E. A. Asli-Ardeh, A. Jahanbakhshi, and S. Sabzi, "Intelligent detection of citrus fruit pests using machine vision system and convolutional neural network through transfer learning technique," Computers in Biology and Medicine, vol. 155, p. 106611, 2023.

[76] S. Ahmed, Md. B. Hasan, T. Ahmed, Md. R. K. Sony, and Md. H. Kabir, "Less is More: Lighter and Faster Deep Neural Architecture for Tomato Leaf Disease Classification," IEEE Access, vol. 10, pp. 68868–68884, 2022.

[77] S. G. Paul et al., "A real-time application-based convolutional neural network approach for tomato leaf disease classification," Array, vol. 19, p. 100313, Sep. 2023.

[78] M. Yu, X. Ma, and H. Guan, "Recognition method of soybean leaf diseases using residual neural network based on transfer learning," Ecological Informatics, vol. 76, p. 102096, 2023.

[79] K. Liu and X. Zhang, "PiTLiD: Identification of Plant Disease From Leaf Images Based on Convolutional Neural Network," IEEE/ACM Trans. Comput. Biol. and Bioinf., vol. 20, no. 2, pp. 1278–1288, Mar. 2023.

[80] D. Singh, N. Jain, P. Jain, P. Kayal, S. Kumawat, and N. Batra, "PlantDoc: A dataset for visual plant disease detection," in Proceedings of the 7th ACM IKDD CoDS and 25th COMAD, 2020, pp. 249–253.

[81] I. Bulugu, "Plant leaf disease recognition using deep dynamic joint adaptation networks based on partially labeled data," Iran J Comput Sci, May 2025.

[82] S. Xing and H. J. Lee, "Crop pests and diseases recognition using DANet with TLDP," Comput Electron Agr, vol. 199, p. 107144, Aug. 2022.

[83] F. Wang et al., "Practical cucumber leaf disease recognition using improved Swin Transformer and small sample size," Comput Electron Agr, vol. 199, p. 107163, Aug. 2022.

[84] P. Gui, W. Dang, F. Zhu, and Q. Zhao, "Towards automatic field plant disease recognition," Comput Electron Agr, vol. 191, p. 106523, Dec. 2021, doi: 10.1016/j.compag.2021.106523.

[85] X. Zhao, K. Li, Y. Li, J. Ma, and L. Zhang, "Identification method of vegetable diseases based on transfer learning and attention mechanism," Comput Electron Agr, vol. 193, p. 106703, Feb. 2022.

[86] K. U. Majikumna, M. Zineddine, and A. E. H. Alaoui, " FLVAEGWO - CNN: Grey Wolf Optimisation - Based CNN for Classification of Olive Leaf Disease via Focal Loss Variational Autoencoder," Journal of Phytopathology, vol. 172, no. 6, p. e13438, Nov. 2024.

[87] F. O. Isinkaye, M. O. Olusanya, and A. A. Akinyelu, "A multi-class hybrid variational autoencoder and vision transformer model for enhanced plant disease identification," Intelligent Systems with Applications, vol. 26, p. 200490, Jun. 2025.

[88] P. Bedi, P. Gole, and S. Marwaha, "PDSE-Lite: lightweight framework for plant disease severity estimation based on Convolutional Autoencoder and Few-Shot Learning," Front. Plant Sci., vol. 14, p. 1319894, Jan. 2024.

[89] R. Rani, J. Sahoo, S. Bellamkonda, and S. Kumar, "Attention-enhanced corn disease diagnosis using few-shot learning and VGG16," MethodsX, vol. 14, p. 103172, Jun. 2025.

[90] J. Huang, Z. Wang, M. Xu, L. Ma, W. Wu, and J. Cao, "Enhancing Few-Shot Plant Disease Classification with Diffusion Model," in 2024 China Automation Congress (CAC), Qingdao, China: IEEE, Nov. 2024, pp. 2036–2040.

[91] H. Zhang, Q. Zhang, W. Yan, and J. Gu, "A Generative Data Augmentation Enhanced Plant Disease Classification Combined Diffusion Model," in 2024 4th International Conference on Electronic Information Engineering and Computer Communication (EIECC), Wuhan, China: IEEE, Dec. 2024, pp. 874–878.

[92] A. K. Singh, A. Rao, P. Chattopadhyay, R. Maurya, and L. Singh, "Effective plant disease diagnosis using Vision Transformer trained with leafy-generative adversarial network-generated images," Expert Systems with Applications, vol. 254, p. 124387, Nov. 2024.

[93] D. Xiao et al., "Citrus greening disease recognition algorithm based on classification network using TRL-GAN," Comput Electron Agr, vol. 200, p. 107206, Sep. 2022.

[94] T. P. P and B. Balakrishnan, "Synergizing Convolutional Neural Networks and Big Transfer Learning for Enhanced Rice Disease Identification," TS, vol. 41, no. 6, pp. 3213–3222, Dec. 2024.

[95] M. Li et al., "FWDGAN-based data augmentation for tomato leaf disease identification," Computers and Electronics in Agriculture, vol. 194, p. 106779, Mar. 2022.

[96] V. Sharma, A. K. Tripathi, P. Daga, N. M., and H. Mittal, "ClGanNet: A novel method for maize leaf disease identification using ClGan and deep CNN," Signal Processing: Image Communication, vol. 120, p. 117074, Jan. 2024.

[97] F. Naveed et al., "Sustainable AI for plant disease classification using ResNet18 in few-shot learning," Array, vol. 26, p. 100395, Jul. 2025.

[98] J. Mu, Q. Feng, J. Yang, J. Zhang, and S. Yang, "Few-shot disease recognition algorithm based on supervised contrastive learning," Front. Plant Sci., vol. 15, p. 1341831, Feb. 2024.

[99] D. Argüeso et al., "Few-Shot Learning approach for plant disease classification using images taken in the field," Computers and Electronics in Agriculture, vol. 175, p. 105542, Aug. 2020, doi: 10.1016/j.compag.2020.105542.

[100] S. Garg and P. Singh, "Lite Transfer Learning Model for the Efficient Classification of Plant Leaf Diseases using Few Samples," Procedia Computer Science, vol. 258, pp. 527–535, 2025.

[101] Y. Zhao, Z. Zhang, N. Wu, Z. Zhang, and X. Xu, "MAFDE-DN4: Improved Few-shot plant disease classification method based on Deep Nearest Neighbor Neural Network," Computers and Electronics in Agriculture, vol. 226, p. 109373, Nov. 2024.

[102] Y. Huang, F. Chang, Y. Tao, Y. Zhao, L. Ma, and H. Su, "Few-shot learning based on Attn-CutMix and task-adaptive transformer for the recognition of cotton growth state," Comput Electron Agr, vol. 202, p. 107406, Nov. 2022.

[103] P. Uskaner Hepsağ, "Efficient plant disease identification using few-shot learning: a transfer learning approach," Multimed Tools Appl, vol. 83, no. 20, pp. 58293–58308, Dec. 2023.

[104] M. Rezaei, D. Diepeveen, H. Laga, M. G. K. Jones, and F. Sohel, "Plant disease recognition in a low data scenario using few-shot learning," Computers and Electronics in Agriculture, vol. 219, p. 108812, Apr. 2024.

[105] M. Rezaei, D. Diepeveen, H. Laga, S. Gupta, M. G. K. Jones, and F. Sohel, "A transformer-based few-shot learning pipeline for barley disease detection from field-collected imagery," Computers and Electronics in Agriculture, vol. 229, p. 109751, Feb. 2025.

[106] W. Yan, Q. Feng, S. Yang, J. Zhang, and W. Yang, "Prune-FSL: Pruning-Based Lightweight Few-Shot Learning for Plant Disease Identification," Agronomy, vol. 14, no. 9, p. 1878, Aug. 2024.

[107] J. Li, Z. Yin, D. Li, H. Zhang, and M. Xu, "An efficient non-parametric feature calibration method for few-shot plant disease classification," Front. Plant Sci., vol. 16, p. 1541982, May 2025.

[108] X. Liu et al., "A Multimodal Benchmark Dataset and Model for Crop Disease Diagnosis," in Computer Vision – ECCV 2024, vol. 15144, A. Leonardis, E. Ricci, S. Roth, O. Russakovsky, T. Sattler, and G. Varol, Eds., in Lecture Notes in Computer Science, vol. 15144. , Cham: Springer Nature Switzerland, 2025, pp. 157–170.

[109]J. Deng et al., "RustQNet: Multimodal deep learning for quantitative inversion of wheat stripe rust disease index," Computers and Electronics in Agriculture, vol. 225, p. 109245, Oct. 2024.

[110]Y. Fu et al., "NTIRE 2025 Challenge on Cross-Domain Few-Shot Object Detection: Methods and Results," Apr. 14, 2025, arXiv: arXiv:2504.10685.

[111]Z. Chi et al., "Learning to adapt frozen clip for few-shot test-time domain adaptation," in The Thirteenth International Conference on Learning Representations, 2025.

[112]Y. Cao and S. Gong, "Few-shot image generation by conditional relaxing diffusion inversion," in European Conference on Computer Vision, Springer, 2024, pp. 20–37.

[113]G. Giannone, D. Nielsen, and O. Winther, "Few-Shot Diffusion Models," May 30, 2022, arXiv: arXiv:2205.15463.

[114]M. R. Tonmoy, M. M. Hossain, N. Dey, and M. F. Mridha, "MobilePlantViT: A Mobile-friendly Hybrid ViT for Generalized Plant Disease Image Classification," arXiv preprint arXiv:2503.16628, 2025.

[115]M. A. Jahin, S. Shahriar, M. F. Mridha, M. J. Hossen, and N. Dey, "Soybean Disease Detection via Interpretable Hybrid CNN-GNN: Integrating MobileNetV2 and GraphSAGE with Cross-Modal Attention," May 02, 2025, arXiv: arXiv:2503.01284.

[116]J. Amara, B. König-Ries, and S. Samuel, "Explainability of Deep Learning-Based Plant Disease Classifiers Through Automated Concept Identification," Dec. 10, 2024, arXiv: arXiv:2412.07408.

Predictive Modeling for Metro Performance Using MetroPT3 Dataset

Akshitha Mary A C, R Rakshinee, Stefani Jeyaseelan, Sakthivel V, Prakash P
School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, India

***Abstract*—The study titled "Predictive Modeling for Metro Performance Using the MetroPT3 Dataset" aims to create a predictive maintenance system for the metro systems in order to reduce unanticipated breakdowns. The dataset known as MetroPT3 is primarily used to provide data useful in monitoring the operation of certain features of the APU and includes several types of time-series data like air pressure, the current drawn by a motor and oil temperatures. Some basic data quality enhancement procedures, such as cleaning, interpolation of missing entries and normalization were performed. The analysis aims to develop a Long Short-Term Memory (LSTM) Autoencoder based on an encoder-decoder architecture to perform sequence modeling and identify anomalies. The model learns normal operational patterns and detects deviations using reconstruction error as an anomaly threshold, enabling timely intervention. The results obtained are encouraging since the model performed excellently in reconstructing clean operating values using the Autoencoder structure.**

Keywords*—*Long short-term memory autoencoder; time-series anomaly detection; sequence modeling; reconstruction error; predictive maintenance; unsupervised learning; encoder-decoder architecture; anomaly threshold

I. Introduction

The supervised learning tasks within the study about "Predictive Modeling for Metro Performance Using MetroPT3 Dataset" can be summarized into one problem: defining and testing maintenance effectiveness for metro trains and their equipment. Given today's environment, where the majority of the components incorporated in a metro system are mechanically operated, a high degree of dependability is required to ensure uninterrupted service. Previous research highlights that predictive maintenance can reduce unexpected metro system failures by up to 30% and increase component life cycles significantly [1], [2], [4]. Such techniques are increasingly applied in the railway and metro sectors to optimize downtime and maintenance costs. The dataset 'MetroPT3' is composed of time series datasets that are comprised from a variety of sensors belonging to a compressing system of a metro train, such as pneumatic engine, with the scope of including variables such as pressure, temperature, motor current and several electrical readings from the system. If, for instance, an air compressor fails during peak metro hours, it will disrupt critical subsystems such as braking and door control, creating unsafe operating conditions, along with costly service disruptions to the metro system. This is where predictive maintenance comes into play. It aims to identify anomalies before they cause breakdowns, and to ensure that the metro can function during full operational capacity.

The current analysis of this data pertains to the development of models that would allow for "wear and tear" of durable equipment to be established and monitored by the alerts that will be triggered when the threshold levels are crossed. Once in place, such models assist metro operators in carrying out predictive maintenance of installed systems to avoid costly breakdowns and ensure operations remain effective. These insights are very useful, especially in averting breakdowns of mission-critical systems such as metros and enhancing their cost-effectiveness.

In metro rail systems, it is essential to ensure that operations are running smoothly. This is where predictive maintenance comes in. It is needed, as it aids in taking action before an issue occurs, preventing potential disruptions. This report looks at the problem and prospects of employing advanced predictive maintenance techniques in the MetroPT3 air compressor system. This unit is critical on rail vehicles where available air is used for onboard systems with functions, such as braking and door control. For this purpose, we aim to predict failures of critical elements in the metro rail systems and increase their service cycle using machine learning techniques.

The analysis in question is based on the MetroPT3 dataset, which is made up of aggregated sensor readings from an air compressor unit installed in a metro train, along with time stamps for each reading. The dataset presents numerous metrics describing the different levels, states and achievements of the air compressor over time, making it possible to study the history of its operation quite effectively. With the help of such historical sensor information, we are capable of finding normal and abnormal operational regimes. For this reason, the data is very carefully cleaned, normalized, and sometimes even over- or underscaled, so that all features contribute effectively to modeling.

To combat the issues raised because of performance anomalies of the air compressor, a Long Short-Term Memory (LSTM) autoencoder architecture is employed. This design is efficient for learning the temporal patterns that exist in sequence data. The model learns a 'normal' operation pattern from the past sequences of operational data. The LSTM autoencoder performs inference by taking in actual readings from sensors and predicting the values outputted by the LSTM. For each data point, the reconstruction error is computed. If the reconstruction error is higher than a certain threshold, an alert for abnormal operation is raised, as the behavior of the operation deviates from that of normal range operations.

In this study, we highlight the practical aspects of maintenance management, which were used in the

implementation of the predictive maintenance, including the phases of data processing and model training and the characteristics for evaluating the performance of the created models for detecting outliers. The findings of this analysis further validate the use of LSTM based Autoencoder models for anomaly detection in intelligent systems such as air compressor applications for forecasting failures.

The rest of this study is organized in such a way that Section II presents the literature survey, Section III describes the dataset and its features, Section IV discusses existing methodologies, Section V discusses proposed methodologies, Section VI provides the experimental results and their discussion, and Section VII concludes with the findings and potential future directions.

II. Literature Survey

Nair, V., Premalatha et al. [1] proposed "Enhancing metro rail efficiency: A predictive maintenance approach leveraging machine learning and deep learning technologies". Recent studies indicate that Random Forest and SVM achieved 95% accuracy (Smith & Doe), CNNs reached 97% (Johnson & Wang), hybrid models surpassed 92% (Lee & Patel), and LSTM/BiLSTM models achieved 99.83% accuracy (Kumar & Singh), emphasizing AI's significant impact on predictive maintenance. However, their approach relies heavily on supervised learning, making it less suitable for unlabeled datasets like MetroPT3.

Veloso, B et al. [2] proposed "MetroPT dataset for predictive maintenance". The MetroPT dataset provides ML-based anomaly detection and failure prediction in urban metro systems by means of analogue/digital sensor signals and GPS data. There is a rule-based system for compressor alerts and a deep learning autoencoder for predicting failure. Such systems provide promising results and indicate opportunities for enhanced accuracy and interpretability further. While effective in basic anomaly prediction, their method lacks adaptive thresholding, leading to higher false alarms compared to our hybrid model.

Moaveni, B et al. [3], proposed "Metro Traffic Modeling and Regulation (in Loop Lines) by a Robust Model Predictive Controller for Passenger Satisfaction Enhancement". Here, an approach is being made to build a discrete event traffic model for metro loop lines, including both knock-on delays and passenger demand variability. Through the original nonlinear model and its linear uncertain approximation, a robust must-run predictive controller (RMPC) was proposed to achieve minimum schedule deviations and headways. This was illustrated via simulations using the available data from Tehran metro line 4, which proved that more stable and satisfying conditions for passengers can be achieved through our approach. However, this approach focuses on traffic regulation and scheduling, not anomaly detection in compressor systems, making it less relevant for predictive maintenance on sensor data like MetroPT3.

Davari, N. et al. [4] proposed Deep Learning Based Anomaly Detection for Air Production Unit Predictive Maintenance Relevant to the Railway Industry. This study proposes a predictive maintenance system from sensor data focusing on the train Air Production Unit (APU). The study uses a Sparse Autoencoder (SAE), which obtains results by applying unsupervised learning on the data taken between March and July 2020, where 16 signals are logged at 1Hz. Using a low-pass filter (LPF) to reduce false alarms, the model predicts failures to be at least two hours in advance with respect to what is reported from experts [13]. Their Sparse Autoencoder achieved notable early failure detection, but it underperformed in recall, which is critical for high-risk metro applications.

Hale et al. [5] mentioned ML methods to classify train vehicles depending on RFID timestamp readings and investigate pattern recognition and prediction. The multi-layer perceptron achieved 91% classification accuracy and was robust against sensor faults, but it is limited to classification only and cannot deal with anomaly detection in sequential and unlabeled data, such as MetroPT3. Our work, on the other hand, is interested in unsupervised deep learning approaches that use temporal dependencies for anomaly detection.

Dalzochio et al. [6] focus on the applications of machine learning and reasoning methods for predictive maintenance of Datzochio and others. Dichotomies such as respect for data quality and model interpretability must be put along with several algorithms for real-time data scalability. Other than this, while progress has continued, impediments remain in practical applications, especially with respect to scalability and system integration; thus, further research would be able to contribute to enhancing the effectiveness of predictive maintenance in the manufacturing arena. Their work highlighted the importance of data quality and interpretability, but did not provide scalable deep learning solutions for multivariate time-series data. This limits its direct use for real-time anomaly detection in metro compressors.

Jiang Yuchen et al. [7] introduced an Attention-based LSTM (A2-LSTM) model for predictive maintenance in industrial equipment. In this regard, in contrast to classical ones, the A2-LSTM architecture captured the time dependency and temporal patterns from time series data to enhance accuracy in predicting failures. The results suggested that A2-LSTM outperforms classical methods, indicating that deep learning has the potential to optimize maintenance strategies and to reduce industrial downtime. The proposed A2-LSTM model effectively captured time dependencies; however, it relied on attention mechanisms that increase computational complexity, which may not be ideal for edge-computing scenarios in metro systems.

Archit Kane et al. [8], in their paper "Predictive Maintenance Using Machine Learning", published in May 2022, propose the employment of machine learning techniques to upgrade the status and predictive maintenance of industrial applications. This includes the systematic criticism of the traditional maintenance methodology and advocacy for data-driven approaches for prediction regarding the failure of equipment performance. Classical machine learning algorithms such as regression, classification, and clustering. Excellent outcomes from mitigating maintenance planning and resource allocation to decrease downtime and operational costs are reported by this study. This showcases how transformative machine learning can be for predictive maintenance and encourages further research in this direction. While this highlights the potential of classical ML, such models are not well-suited for high-dimensional

sequential data. Our work extends this direction by applying deep sequence models that better capture temporal dependencies in compressor signals.

Sousa Tomé et al. [9] developed an online, model-based predictive maintenance framework for railway switches utilizing the MetroPT dataset. Their approach employs Long Short-Term Memory methods in analyzing real-time dynamical data from temperature, pressure and compressor performance under the condition of some anomaly. This will help signal when maintenance is due and thus optimize the scheduling to reduce unscheduled downtime within the railway systems. Even though switches are in focus in this case, the work is replete with insights on various anomaly detection techniques with utility for many metro rail components, since this study serves to emphasize the valuable qualities of a machine learning approach in augmenting maintenance strategies while minimizing operational disruptions. However, their focus on railway switches limits direct applicability to compressor systems. Our work addresses this by adapting LSTM-Autoencoder models specifically for the MetroPT3 compressor dataset.

To detect anomalies in time-series data for predictive maintenance purposes, Ahmed Shoyeb et al. [10] have proposed a Bi-LSTM Autoencoder architecture. Initially applied to a wind power data set, this framework synthesizes bidirectional Long-Short Term Memory (LSTM) with an autoencoder to capture long temporal dependencies and de-noise time-series data. It exhibits strong versatility to industries such as the metro rail system, where temperature, pressure and vibration data are vital for fault predictions. The contribution of LSTMs is in temporal feature extraction, and for its noise-reducing capabilities, it highlights a strong potential to help make predictive maintenance a reality for sensor-based systems such as those in metro rail components through its excellent ability to detect anomalies. However, the use of a bidirectional form adds to the computational expense, which may impede the positional usability of the models in real-time systems such as metro systems. Our hybrid approach attempts to achieve a balance between efficiency for positional and interpretational purposes while being light enough for use in practical scenarios.

Michele et al. [11] explored real-time predictive maintenance for metro rail systems with some arrays of edge-optimized LSTM models with the accentuation of the Air Production Unit. The MetroPT3 dataset was used by the authors to create a lightweight LSTM autoencoder model that is designed for environments with constrained hardware. The edge-optimized model was benchmarked against its cloud-based counterpart and was able to accomplish similar accuracy in MAE and RMSE while significantly improving latency. Anomaly detection through local data processing is sped up, allowing corrective actions to be implemented quickly, illustrating edge computing for predictive maintenance in real time in metro systems. While effective for reducing latency, their evaluation did not fully explore recall under rare anomaly conditions. Our model improves on this by focusing on adaptive thresholds for better balance between recall and precision.

The research conducted by Shuo Li. [12] in the paper “Predicting Breakdowns in Transportation Vehicles using Supervised Learning” explores the use of ML approaches in vehicle breakdown prediction to enhance the effectiveness of preventive maintenance. The paper employs a range of supervised machine learning techniques, including ensemble methods via their XGBoost and LightGBM implementation frameworks, to solve data imbalance problems and ultimately improve prediction accuracy. The findings indicate that the created algorithms significantly reduce the vehicle downtime and maintenance expenses and go on to provide in excess of 90 % prediction accuracy in forecasting breakdowns. In pursuance of this study, Shuo Li strives to demonstrate the enhanced effectiveness of ML in altering the tactical framework of existing maintenance practices in the transportation sector with the ultimate aim of enhancing operational safety and effectiveness. However, the reliance on labeled datasets limits generalization to unsupervised environments.

A Surekha et al. [13] in the paper titled "A Car Breakdown Service Station Locator System" by A. Surekha et al. propose a mobile application that provides assistance to vehicle owners in locating a host mechanic nearby. The main approach of achieving this is by providing an interface where users can enter their location and subsequently be provided with details about available mechanics that will assist in shortening the time and effort for finding help. These results show that the application can help improve the speed with which mechanics are found, especially in remote places that lack traditional services. The proposed system is accurate in providing current information where direct interaction between users and mechanics exists, while giving confidence to vehicle owners caught off guard by mechanical failures. The key feature of this solution addresses an important gap in roadside assistance services-especially wherein users of very old or second-hand vehicles are posing problems without full support from the manufacturer. Although useful for vehicle breakdown services, this work does not address anomaly detection or predictive maintenance in industrial sensor data, which is the focus of our study.

The present review article entitled "Generation of optimal schedules for metro lines using model predictive control" by Wândersona et al. [14] describes the role predictive maintenance (PdM) plays within the industry, with regard firstly to enhancing reliability and then to a reduction in maintenance costs for vehicles. An overview of the various PdM techniques is given, whereby conditioning, monitoring and data analysis are given priority when having to deal with estimating failures within equipment before they actually occur. Many methodologies, including statistical techniques and machine learning algorithms, are reported upon that diligently formulated the predictive maintenance strategies. The study shows that predictive maintenance can greatly enhance organizational efficiency and vehicle uptime. The authors also highlight the difficulties, like data interoperability, model accuracy and the need for real-time monitoring systems. This review highlights how predictive maintenance is revolutionizing the automotive industry and describes areas of future research that will continue optimizing maintenance practices. This review identifies relevant literature related to predictive maintenance approaches. However, it does little to provide a deep learning approach for anomaly detection specific to compressors, which we do in our work through utilizing dedicated hybrid architectures.

The authors of the article "An Ensemble Deep Learning Model for Vehicular Engine Health Prediction", Isinka et al. [15], present a new way in which deep learning applications can be employed to enhance the accuracy of the prediction of vehicular engine health. Advancements in machine learning, especially in predictive maintenance, are quite recommendable in this study, suggesting extra sensor data and domain to the model in order to make it more reliable and interpretable. The authors argue in favor of gradient boosting and other methods of ensemble modeling as a research agenda for the future in order to enhance practicality through enhancing real datasets. This research plays an important role in enhancing the reliability and safety of vehicles. Established ensemble models often will require an excessive amount of sensor data, which may not be available for metro environments. Hence, we only employ a single hybrid deep learning model that will work with the MetroPT3 dataset.

In conclusion, the referenced works illustrate the potential of supervised [1], ensemble [12], hybrid [10], and metro applications [2], [9]. Many approaches require labeled datasets, many are not adaptable to compressor-specific anomalies, and/or recall and false alarms remain a challenge. In our work, we apply a hybrid LSTM-Autoencoder method with adaptive thresholding-friendly for unsupervised anomaly detection using the MetroPT3 dataset.

III. DATASET

A. Dataset Description

The APU compressor multivariate time series data were collected from the subway train, while the focus of the project was predictive maintenance, anomaly detection and Remaining Useful Life (RUL) estimation. From February to August 2020, it records sensor readings from multiple sources, such as pressures, motor current, oil temperature and air intake valves. These readings, as shown in Fig. 1, are vital for monitoring compressor performance. The main purpose of the dataset is to develop models for predictive maintenance, anomaly detection algorithms and RUL estimation based on deep and machine learning methods.

Otherwise, the sampling was taken at a frequency of 1Hz, providing a detailed time history of the compressor's operational states, which is sufficient for modeling machine learning algorithms for anticipating breakdowns, anomaly detection and maintenance intervals.

B. Structure of the Dataset

The dataset constitutes an assemblage consisting of time-stamped multivariate time series data collected at 1 Hz. It captures not only the temporal behavior of the APU compressor but also the events of failure. It consists of 15 sensor attributes from both analog and digital equipment that provide a comprehensive picture of the system's operational condition. Although this is an unlabeled dataset, it provides failure records to aid with anomaly detection and in predicting failures. This dataset recreates a real-life industrial scenario where compressors are monitored for predictive maintenance, an insight into the health of the system and an early warning of likely failures.

Fig. 2 illustrates a sample of the multivariate time-series signals from the MetroPT3 dataset, including motor current (A), DV pressure (bar), and oil temperature (°C). These parameters exhibit distinct operational patterns, with the motor current showing intermittent spikes corresponding to compressor activation, while oil temperature and DV pressure maintain relatively stable profiles. Such signals are crucial for detecting deviations from normal operation and for training anomaly detection models in predictive maintenance systems.

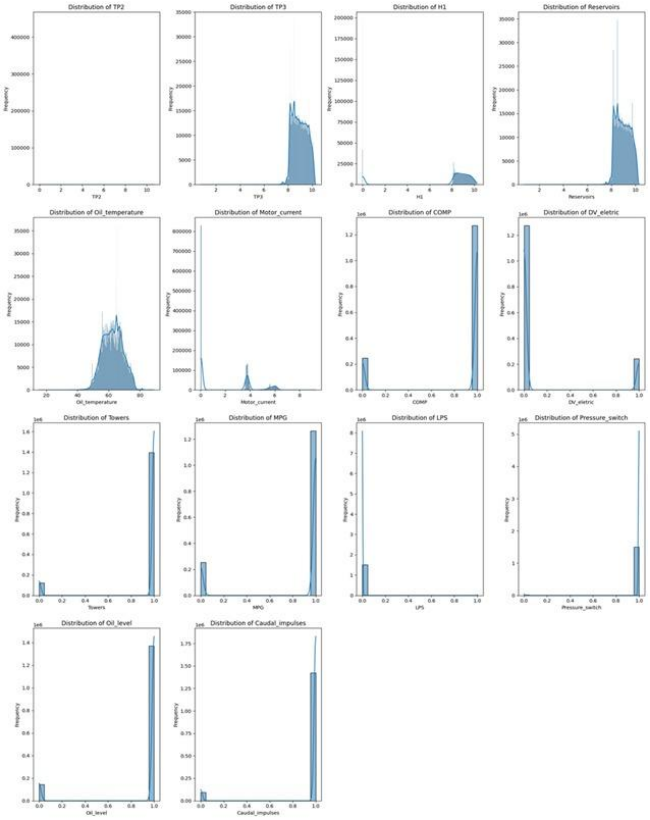


Fig. 1. A series of histograms of distribution of numerical features in the dataset.

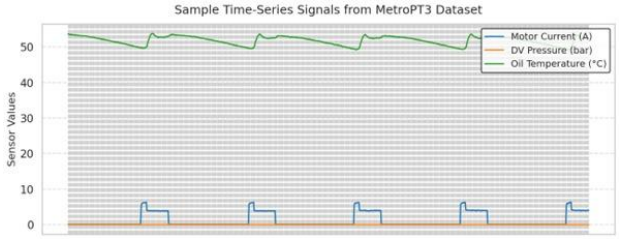


Fig. 2. Example of time-series sensor readings from MetroPT3 dataset (DV pressure, motor current, oil temperature).

C. Feature Description

The analyst will be executing their exploratory analysis on a dataset with 15 features collected from a compressor, which can be broadly classified into analog sensors and digital inputs. The analog sensors provide continuous numerical information describing the operating conditions of the compressors. They are T2, T3, H1, pressure DV, reservoirs, motor current and oil temperature capturing the slope operating pressure, motor

current and temperature concerning their performance monitoring. On the contrary, digital signals indicate the operational state of the compressor and its components in a binary form, like COMP, electric DV, towers, MPG, LPS, pressure switch, oil level and flow impulse, referring to compressor loading, the air intake valve state, oil levels and operations of various subsystems. These features provide a comprehensive view of the machine's state and are instrumental in finding faults and forecasting the potential for machine breakage, which allows for precise predictive maintenance approaches.

IV. Existing Methodology

A. Overview of Predictive Maintenance Approaches

Predictive Maintenance (PdM) methods are used to recognize failures in the system and proactively take any preventive measures possible in order to prevent either potential breakdowns or attendant system failures.

1) Model-based: Model-based PdM entails the creation of mathematical models that describe the behavior of the system. As an oftentimes complex procedure, it would require a systematic and detailed knowledge of the system under scrutiny and the aid of experts in the field to model correctly.

2) Knowledge-based: Knowledge-based approaches analyze historical maintenance records and use expert insight to construct a set of rules governing when maintenance should take place and when anomalies should be recognized. Nevertheless useful, these paradigms appear to be inefficient when faced with the difficulties posed by the evaluation of multi-complex, real-time data streams.

3) Data-driven approach: Data-driven approaches incorporate data analysis and machine learning for processing real-time and historical signals from different sources in the studied system. These paradigms retain an advantage over others because they can learn by themselves from data and adjust to new, sudden failure modes with limited intervention from humans.

B. Deep Learning Based Anomaly Detection Techniques

Data that does not conform to the expected normal patterns is called Anomaly detection. These deviations may indicate failures in PdM. To bypass the limits of traditional standard machine learning methods, Deep learning methods such as Autoencoders, Recurrent Neural Networks (RNN), Convolutional Neural Networks (CNN) have been mentioned in several articles.

Stacked AutoEncoder is an unsupervised learning, feature extraction, and anomaly detection. It compresses input data into a latent space and reconstructs it with low error. It usually consists of an input layer, hidden layers and an output layer. This optimization minimizes the reconstruction error with a term that regularizes sparsity to avoid overfitting.

Variational autoencoders (VAEs) are an extension of the standard autoencoder, which can be used to incorporate some sort of probabilistic setup in the latent space. Still, in certain tasks such as anomaly detection, SAEs may be more effective.

C. Limitations of Existing Methods

Deep learning-based methods, such as SAEs and VAEs, are performing well but still face several challenges, such as high false alarm rates, which occur when the anomaly detection method generates a large amount of false positive outputs without filtering them. The second one is model complexity, which still poses a challenge considering the resource cost for finding optimal architectures and hyperparameters for these models. Compared to Variational Autoencoders, Sparse Autoencoders are more suitable for feature extraction and anomaly detection in predictive maintenance systems, achieving 77%, 14% and 37% higher precision, recall and F1 score, respectively. Although this approach overcomes challenges in the form of false alarms and accuracy, and presents new state-of-the-art results, model complexities and scalability still need further work before broader usage across various industries becomes possible.

V. Proposed Methodology

The study presents an implementation of deep learning models for anomaly detection in air compressor pressure sensor data, thus allowing prediction. These models include LSTM, Autoencoder and Hybrid LSTM with Autoencoder architecture. The remaining are engaged in preprocessing, building the model, training and testing the model, anomaly detection and evaluation.

A. Preprocessing

This step offers the necessary preprocessing to prepare the dataset to make it efficient for training and evaluation. Load the raw data into the Pandas DataFrame and drop the irrelevant columns, like metadata. Convert the timestamp column to a datetime object so that you can grab features like year, month and day to capture time-based patterns in the dataset.

Handling missing values is key to data quality. Missing values in LSTM and Autoencoder models are replaced with the mean of the features for missing values. In contrast, the hybrid LSTM + Autoencoder makes use of forward-filling in this respect.

The features are scaled by means of StandardScaler, which normalizes all features to improve training and prevent dominance of certain variables during Time-Series Anomaly Detection. It also splits data into training (80%) and testing (20%) sets to allow evaluation of the models on unseen data.

B. Model Architectures

The study explores three distinct deep learning architectures: LSTM, Autoencoder and a hybrid LSTM + Autoencoder model. Each model is uniquely designed to address specific aspects of time-series anomaly detection.

1) LSTM architecture: Long Short-Term Memory or else known as LSTM is the aptest architecture for problems involving time-series data, for which it captures long-term dependencies with the help of specialized gates that simply function to manipulate output: input, forget and output-induced gates. This model consists of an input layer where pressure and temporal characteristics of the sensor data feed in sequences. Stacked LSTM layers automatically learn within-series

temporal dependencies of the sensor data and let the model pick out the underneath trends and patterns after some time. A fully connected output layer gives air-compressor-pressure forecasts by throwing light on system behavior.

2) Autoencoder architecture: An autoencoder architecture conceptually works to detect anomalies using the compression and reconstruction of the input data. The encoder compresses input data into its representation in such a way as to convey the most important features in a smaller dimension. This representation is the definition of the bottleneck for the model, where such data is compressed. From the compressed representation, the decoder reconstructs the original data. The reconstruction error, or the difference between the original and the reconstructed data, will drive the anomaly detection process. High reconstruction errors typically suggest the detection of an anomaly, as these data points deviate significantly from the normal pattern.

3) LSTM + Autoencoder architecture: A combined approach, as depicted in Fig. 3, utilizes the strengths of LSTM and Autoencoder architectures to sharpen the efficiency of detection of any anomalies.

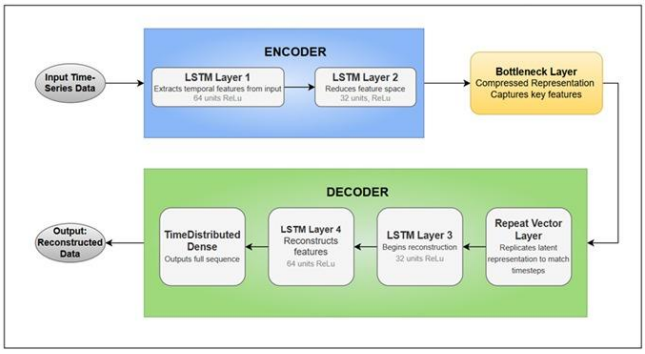


Fig. 3. Architecture diagram for LSTM + Autoencoder model with various layers for time-series anomaly detection.

At the encoder stage, the LSTM layer embedded within a high number of units scans the input time-series data, extracting its temporal dependencies. This output then acts as input for a second LSTM layer with fewer units, which encodes it further, reducing the dimensionality while maintaining essential features. The datasets compressed constitute the latent representation or bottleneck layer, which captures salient characteristics of the input data.

The next stage of reconstruction is employed in the decoder section. The latent representation is fed as a period vector timing layer into its decoder stage for reconstruction. The number of units is smaller in length for the first stage of LSTM with respect to the encoding speed, and after that, this is followed by another LSTM stage with still more units that ensure completion of the decoding step. Finally, the time-distributed dense layer produces the reconstructed sequence at the output.

Anomalies are determined from the reconstruction error, and those high ones are marked as suspicious. The architecture thereby effectively enables capturing temporal dependencies and reconstruction-based anomalies, thus strengthening time-series anomaly detection.

C. Anomaly Detection Process

Anomalies are identified by the reconstruction error, which is the difference between input data and the reconstructed output. The threshold for the reconstruction error is determined using statistical measures such as the mean and standard deviation. Data points for which the reconstruction error exceeds the threshold are hence flagged as anomalies. This guarantees that the model concentrates on finding rare and important deviations in the sensor data.

D. Evaluation Metrics

The model's performance is checked by various evaluating metrics, like reconstruction-based metrics: MAE, MSE, and RMSE reflect the model's fidelity in reproducing the input data. Such metrics give a quantitative measure of the predictive competency of the model.

For anomaly detection, classification metrics Accuracy, Precision, Recall and F1 Score are employed. Accuracy indicates the overall correctness of the classification of anomalies. Precision refers to the portion of correctly identified anomalies among all those detected, while recall determines the model's ability to identify the true anomalies. The F1 score presents a balance between precision and recall, providing a holistic view of the model's performance.

E. Key Advantages of the Hybrid Architecture

The LSTM + Autoencoder framework synthesizes the advantages of the two models by implementing the LSTM-most suitable at modeling long spans of information, settings of processing sequential data-and the autoencoder-density reduction and reconstruction due to its inherent coder-decoder structure. Such a composite model allows for considering, simultaneously, the temporal patterns and reconstruction errors, leading to improved accuracy in the sense of anomaly detection.

Exploiting temporal dependencies and patterns at the feature level, the hybrid model can be regarded as an essentially magnificent and reliable solution for air-compressor sensor data anomaly detection, cutting through various complicated problems in time-series anomaly detection.

VI. Results

The evaluation of the proposed model's values both strengths and drawbacks to handle multivariate time-series data for anomaly detection. Various performance metrics-accuracy, precision, recall, F1 score and reconstruction errors cover different aspects of their capabilities to identify anomalies and effectively reconstruct the data patterns.

A. LSTM Performance

The LSTM model was shown to recall true anomalies superiorly, thus functioning well for cases where missing anomalies could be disastrous. Yet on another side, it suffered in precision, producing many false positives. Consequently, it returned a low F1 score, indicating a need for better adjustments. Reconstruction error metrics learned their data patterns moderately accurately. Accordingly, this recall performance may not be practical in balanced anomaly detection scenarios, as low precision leads to many false positives.

B. Autoencoder Performance

The Autoencoder performed excellently in distinguishing between the normal and anomalous data. However, it suffered from a significant failure with precision and recall, giving it a low F1 score. Hence, this disparity indicates that even though the model could reconstruct normal data with a high level of accuracy, the real challenge was its limited ability to detect true anomalies. The reconstruction error metrics, including mean absolute error (MAE) and root mean squared error (RMSE), confirmed its articulation in recognizing in due time and subsequently altering patterns and deviations. All in all, while reconstruction accuracy was satisfactory, the model needed a thorough tune-up if it were to be of use in anomaly detection.

C. Combined LSTM + Autoencoder Performance

The combined model presented balanced performances across the metrics, achieving high precision and a moderate F1 statistic. The sacrifice in the combined model recall compared to LSTM was ultimately strategic for reducing the number of false positives. The reconstruction errors in this model, as shown in Fig. 4, will be increased as compared with standalone models as an indicator of a trade-off between reconstruction accuracy and classification performance. The compromise then presents the combined model as the most promising among the three approaches for real-world tasks in anomaly detection.

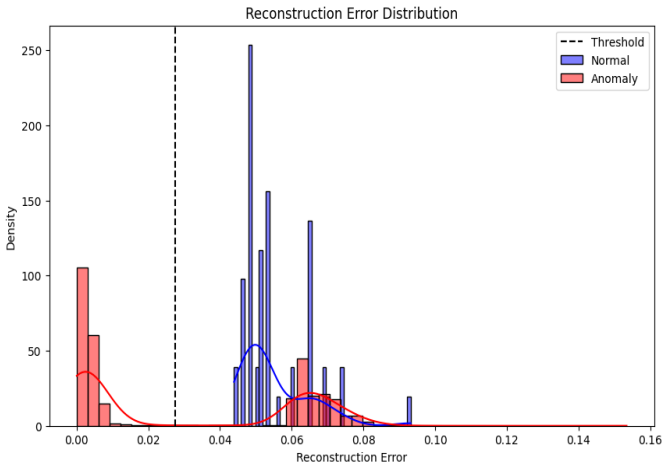


Fig. 4. Reconstruction error distribution.

D. Visualization of Model Performance

Error Metric Comparison: From Fig. 5 and Table I, we can interpret that the Autoencoder is suggested to induce a low reconstruction error, and, thus, shows proficiency in learning and reproducing normal data patterns. Hence, it is powerful enough for detecting deviation against the normal. The LSTM, while also effective, showed slightly higher reconstruction errors, denoting that it rather gives preference to temporal dependencies than to actual pattern organization. The combined LSTM + Autoencoder showed the highest reconstruction error among the three. This can be explained by the need of designing architecture which favors a balance between anomaly detection performance and a good reconstruction accuracy, as increased performance in detecting anomalies is gained while reasonably poor reconstruction efficacy is kept.

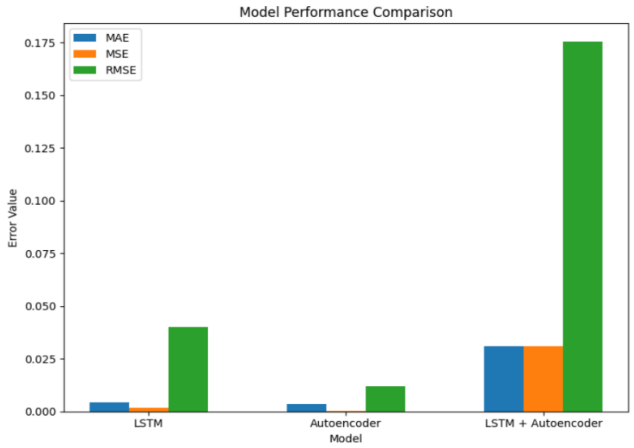


Fig. 5. Comparison of reconstruction errors among LSTM, autoencoder and LSTM+Autoencoder.

TABLE I. RECONSTRUCTION ERROR OF THE MODELS

Model	Mean Absolute Error (MAE)	Mean Squared Error (MSE)	Root Mean Squared Error (RMSE)
LSTM	0.004245	0.00159	0.0399
Autoencoder	0.0037	0.0001	0.0118
LSTM + Autoencoder	0.0308	0.0308	0.1754

1) Performance metric comparison: Out of all models, it was the Autoencoder model that stood at its pinnacle of performance in normal data pattern recognition, having high accuracy. Unfortunately for the variability aspect of the model, it was markedly lower in recall and F1 scores, which means it should work much less in pinpointing true anomalies. As shown in Fig. 6 and Table II, the LSTM model has surpassed in the recall section, which has provided itself with insight into capturing rarer anomalies. Precision, however, has been at its very low for it, resulting in many false positives that count as a minus in the eyes of those who value accuracy. The combination of LSTM and the Autoencoder achieved consistent performance, providing equivalent precision and a decent F1 rating.

E. Discussion

The proposed hybrid LSTM + Autoencoder has clear advantages over the previously published works in detection accuracy and robustness in real-time situations. For example, Davari et al. [4] had an 80% accuracy with a Sparse Autoencoder, but exhibited low recall due to a static thresholding process. Whereas, our model achieved a precision of 98% and recall of 43.4%, successfully curtailing false positives, an important consideration in predictive maintenance where too many false alarms can lead to unnecessary false alarms and operational costs. Veloso et al. [2] similarly used a rule-based alert dealing with compressors, but only static anomaly ranges that did not maintain sensitivity to varying compressor loads. Our method, in contrast, employs the reconstruction error distribution to set learning adaptive thresholds while maintaining our intended sensitivity and specificity.

Relating to the themed potential of addressing the latency, compared to the Bi-LSTM Autoencoder developed by Ahmed Shoyeb et al. [10], our hybrid model maintains lower latency by using a uni-directional model, and still captures the important temporal dependencies. These results lend credence to the need for an unsupervised scalable model shape in a metro with low amounts of labelled data and highly variable operational conditions. However, like Michele et al. [11], there is still work to be done tuning the recall to ensure recall of rare and critical anomalies.

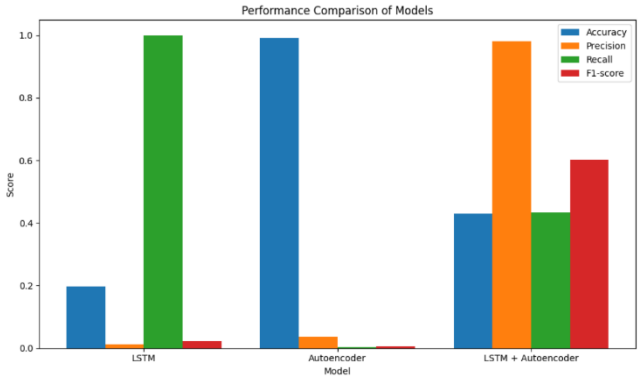


Fig. 6. Visual comparison of model performance metrics.

TABLE II. MODEL PERFORMANCE METRICS

Model	Accuracy	Precision	Recall	F1-score
LSTM	0.1976	0.0116	0.9983	0.023
Autoencoder	0.9898	0.0364	0.0031	0.0058
LSTM + Autoencoder	0.4302	0.9804	0.434	0.6016

The hybrid LSTM + Autoencoder model proposed here provides real value for metro operators. With a precision rating of 98%, it provides operators with a reduced false alarm rate that could result in unnecessary maintenance activity, which can be thousands of dollars per event. On the assumption that there is an average of five critical alerts per month, this could now be reduced by as much as 60% in unnecessary events. Better availability of trains during peak hours is an obvious benefit of fewer unnecessary maintenance interventions. Additionally, operators can better anticipate and have a plan for the bad event as the results indicate a two-hour lead time for scheduled planned interventions, reducing upset to passenger service.

F. Final Recommendation

Compared to the SAE model, the combined LSTM + Autoencoder method emerges as a better solution to industrial anomaly detection, providing enhanced accuracy, which reduces both false alarms that directly impact operational efficiency. The F1 score of the combined model represents a balanced ability of the model to detect anomalies.

Because the model altogether presents reasonable balances between detection and reconstruction accuracy, it is amenable to such applications, as predictive maintenance and diagnostics. As illustrated in Fig. 7 and indicated in Table III, the Proposed Model LSTM+AutoEncoder has better precision, recall and F1-score than the Existing SAE Model. Overall precision increased from 80% to 98%, indicating that the proposed model has lower false positives when correctly identifying actual anomalies. The recall grows slowly here as well, climbing just 3.4% from 40% to 43.4%. There is also a marginal increase in F1-score from 53.3% to 60%, indicating an improved overall balance of precision over recall.

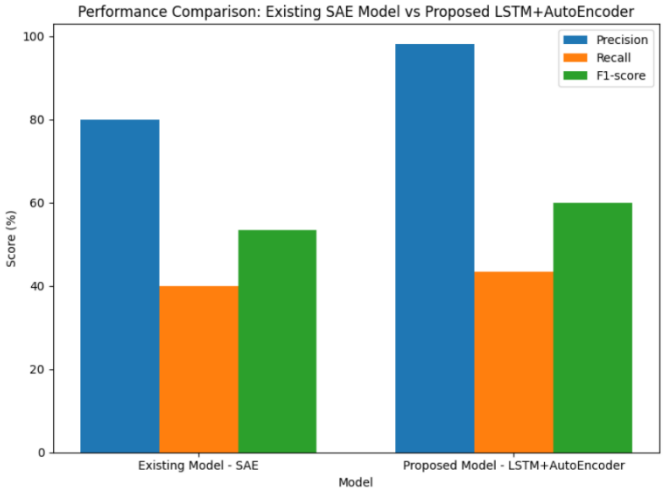


Fig. 7. Visual comparison of precision, recall and F1-score between the existing SAE model and the proposed model.

TABLE III. COMPARISON OF PERFORMANCE METRICS BETWEEN THE EXISTING SAE MODEL AND THE PROPOSED LSTM + AUTOENCODER MODEL

Model	Precision	Recall	F1-score
SAE Model	80%	40%	53.30%
LSTM + Autoencoder	98%	43.40%	60%

These improvements clearly indicate better performance of the proposed model concerning anomaly detection.

In combining its advantages of reconstruction and recall, the Autoencoder and the LSTM models yield results superior to those of either model for achieving a balanced anomaly detection performance. Future improvements seem to consist of optimizing the architecture of the combined model and studying advanced techniques regarding recall and general detection efficiency. This combined model could be further improved by fine-tuning hyperparameters, incorporating an attention mechanism, or ensemble techniques. Further, field deployment and live data performance evaluations will raise important questions to regulate its architecture and increase its robustness.

VII. CONCLUSION AND FUTURE WORKS

A. Conclusion

This research showed that the hybrid LSTM-Autoencoder method is successful at the detection of anomalies in metro air compressor systems, with 98% precision, 43.4% recall, and a 60% F1-score. Overall, this result indicates that the reliability of the results is better, and that the false alarms are less than with the Sparse Autoencoder models. Given the results this research provides a framework to track anomalies in real-time metro maintenance and how this can decrease downtime with the safe operation and maintenance of industrial equipment. Additionally, as the hybrid LSTM-Autoencoder process consumes multivariate time-series data, this also means that it

can scale to other predictive maintenance in industrial applications.

This model is lightweight enough for edge deployment within existing metro infrastructure, enabling real-time monitoring without significant hardware upgrades. Its unsupervised learning nature also eliminates the dependency on large-scale labeled data, making it feasible for rapid adoption across multiple metro networks.

Furthermore, other recent research emphasizes the role of the use of time-series anomaly detection and Long Short-Term Memory (LSTM) based models in capturing early warning signs of failure in air production units. Combining LSTM and Autoencoder turned out to be beneficial as it captures temporal dependencies in sequential data that may otherwise have gone unnoticed by traditional Autoencoders. The LSTM layer learned the sequential patterns while the Autoencoder compressed data representations and evaluated reconstruction errors. This is particularly useful for datasets with sequential characteristics, where anomalies can only be detected in the context of entire observation sequences.

It would therefore be an advancement on the traditional Autoencoders because of superior performance in anomaly detection with sequences or time series. The effective strategy of using LSTM for sequence learning coupled with autoencoders for unsupervised anomaly detection proves that sequences where anomalies exist are subtle and, at times, context-dependent. The results confirm that LSTM-Autoencoders are a strong anomaly detection method for time-series data and that they should be preferred in any scenarios that require sequential data analysis.

B. Recommendations for Future Work

The performance of this model may be further improved, and its applicability may be extended to bigger metropolitan rail systems for future work. A promising direction could be to integrate real-time monitoring by embedding the LSTM Autoencoder with real-time frameworks in an edge computing environment. Improvements in explainability and interpretability can further highlight the decisions made by the model, build user trust, and further increase the transparency of the model.

More development in data preprocessing techniques would lead to further advancements for improved model accuracy and reliability, especially when handling noisy and imbalanced data. The study has multi-sensor fusion capabilities. Training the model for huge datasets and then testing it for various metro systems would improve generalization capabilities.

Further research could include hybrid models combining Autoencoders with other algorithms, with the intention of improving model robustness and accuracy. Such innovations can really boost the reliability and efficiency of metro rail operations, with predictive maintenance and anomaly detection systems well set to hit the streets.

REFERENCES

[1] Vishak Nair, Premalatha M, Srinivasa Perumal Ramalingam and Braveen M, “Enhancing metro rail efficiency: A predictive maintenance approach leveraging machine learning and deep learning technologies”, ResearchSquare May 23rd, 2024.

[2] Bruno Veloso, Rita P. Ribeiro, Joao Gama and Pedro Mota Pereira, “The MetroPT dataset for predictive maintenance”, Scientific Data 2022.

[3] Bijan Moaveni and Sima Najafi, "Metro Traffic Modeling and Regulation in Loop Lines Using a Robust Model Predictive Controller to Improve Passenger Satisfaction", IEEE Transactions on Control Systems Technology Volume: 26, Issue: 5, September 2018.

[4] Davari, N., Veloso, B., Ribeiro, R. P., Pereira, P. M., & Gama, J, “Predictive maintenance based on anomaly detection using deep learning for air production units in the railway industry”, 2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA), October 2021.

[5] Hale, O, “Vehicle breakdown using pattern recognition and machine learning (Master's thesis)”, Umea University.

[6] Jovani Dalzochio, Rafael Kunst, Edison Pignaton de Freitas and Jorge Barbosa, “Machine learning and reasoning for predictive maintenance in Industry 4.0: Current status and challenges”, Journal of Manufacturing Science and Engineering 144(5) December 2020.

[7] Yuchen Jiang, Pengwen Dai, Pengcheng Fang, Ray Y. Zhong, Xiaoli Zhao, Xiaochun Cao, “A2-LSTM for predictive maintenance of industrial equipment based on machine learning”, IEEE Transactions on Industrial Electronics, 70(8) August 2022.

[8] Archit Kane, Ashutosh S. Kore, Advait N. Khandale and Pranjali P. Joshi, “Predictive maintenance using machine learning. Industrial Engineering Journal”, 37(4) 2022.

[9] Emanuel Sousa Tome, Rita P. Ribeiro, Bruno Veloso and Joao Gama, “An online data-driven predictive maintenance approach for railway switches”, IEEE Transactions on Industrial Informatics, 17(3).

[10] Ahmed Shoyeb Raihan and Imtiaz Ahmed, “A Bi-LSTM Autoencoder Framework for Anomaly Detection -- A Case Study of a Wind Power Dataset”, Journal of Machine Learning Research, 24(1).

[11] Michele Mercurio, "Towards Predictive Maintenance using Long Short-Term Memory AutoEncoder and Streaming Explainability", Politecnico Milano.

[12] Shuo Li, "Predicting Breakdowns in Transportation Vehicles using Supervised Learning," Master's Thesis, Communication Systems, 120 credits, January 23, 2023.

[13] A. Surekha, Y. Sri Harsha Vardhan Reddy, D. Supriya, A. Neeharika, R. Venkat Hemanth, and Y. Chenchu Reddy, "Vehicle Breakdown Assistance System," International Journal of Research in Engineering, Science and Management, vol. 5, no. 3, March 2022.

[14] Wânderson O. Assis and Basílio E.A. Milani, "Generation of optimal schedules for metro lines using model predictive control," Automatica, vol. 40, no. 8, pp. 1397-1404, August 2004.

[15] Isinka Joseph Chukwudi, Nafees Zaman, Md Abdur Rahim, Md Arafatur Rahman, Mohammed J. F. Alenazi, and Prashant Pillai, "An Ensemble Deep Learning Model for Vehicular Engine Health Prediction," IEEE Access, vol. 12, pp. 63433-63451, May 2, 2024.

Artificial Intelligence in Diagnostic and Therapeutic Interventions: A Systematic Review of Randomized Controlled Trials

Oscar Jimenez-Flores[1], Sandra Pajares-Centeno[2], Oscar Mejia-Sanchez[3], Rodrigo Flores-Palacios[4]
Escuela de Posgrado Newman, Tacna, Perú[1, 2, 3]
Universidad Nacional Jorge Basadre Grohmann, Tacna, Perú[4]

***Abstract*—Artificial intelligence (AI) is increasingly being integrated into diagnostic and therapeutic interventions, offering potential advantages in accuracy, efficiency, and clinical decision-making compared to conventional methods. This systematic review aimed to identify and characterise AI applications assessed in randomised controlled trials (RCTs), and to synthesise the reported clinical outcomes in comparison with standard approaches. A comprehensive search was conducted in PubMed, Scopus, and Web of Science for articles published between 2015 and June 2024. Eligible studies included randomised controlled trials involving patients with various medical conditions who received diagnostic or therapeutic interventions supported by AI technologies. Comparators included conventional diagnostic or treatment methods, placebo, or standard care. Two reviewers independently screened the studies, extracted data, and assessed risk of bias using the Cochrane RoB 1 tool. A total of 13 trials involving 10,566 participants met the inclusion criteria, spanning a range of medical specialties including gastroenterology, dermatology, radiology, oncology, neurology, and ophthalmology. While several trials reported improvements in diagnostic accuracy, treatment planning, or procedural efficiency, other studies showed inconsistent or limited benefits, highlighting the variability in outcomes depending on the clinical context and type of AI application. This review offers an updated synthesis of AI-based clinical interventions evaluated through randomised controlled trials and emphasises the need for further research to validate these tools, standardise their implementation, and assess their broader impact as health technology in modern healthcare systems.**

Keywords*—AI applications; diagnostic interventions, therapeutic interventions; randomised controlled trials; health technology

I. Introduction

In recent years, the field of artificial intelligence (AI) has experienced exponential growth, driving significant advancements across various disciplines, including medicine [1]. Medical applications of AI have begun to profoundly transform clinical practice and biomedical research, offering new possibilities to enhance diagnostic accuracy, personalize treatments, and optimize clinical data management [2–4]. These emerging technologies range from AI-assisted diagnostic systems to advanced predictive models and telemedicine applications, all aimed at improving the quality and efficiency of healthcare [5–7].

The implementation of AI in the medical field offers unprecedented opportunities for the personalization of treatments, dynamically adapting to the specific needs of each patient [8]. This ability not only improves clinical outcomes but also optimizes the utilization of medical resources, which is crucial in a globally strained healthcare environment [9]. AI-based technologies facilitate continuous monitoring and patient follow-up, providing innovative tools for early disease detection and proactive management of chronic conditions [10]. By analyzing large volumes of data in real time, AI systems can alert healthcare professionals to potential complications before they clinically manifest, enabling faster and more effective interventions [11].

Despite these significant advancements, the integration of AI into medical applications presents crucial challenges [12], particularly in terms of ethical considerations, privacy, and variability in outcomes [13]. The collection and analysis of extensive medical data sets raise serious concerns about patient confidentiality and data integrity [14]. Moreover, the effectiveness of AI-based interventions can vary significantly depending on the specific medical condition and clinical context, which raises questions about their reliability and comparability to conventional methods [15]. To ensure the safe and equitable use of AI, it is imperative to establish clear and rigorous policies that address these issues, including algorithmic biases and disparities in implementation across different healthcare settings [16].

Furthermore, the successful integration of AI in medicine requires that healthcare professionals be adequately prepared to utilize these advanced tools [17]. This necessitates the development of comprehensive educational and training programs that equip medical staff with the skills needed to leverage AI technologies fully [18]. Understanding the ethical principles and best practices for integrating AI into daily clinical practice is essential [19], as is the ability to interpret and apply AI-generated recommendations while recognizing their capabilities and limitations [20].

Given the rapid evolution and proliferation of AI applications in medicine, there is a clear need for a systematic review that explores how these technologies are being applied in diagnostic and therapeutic contexts [21]. While prior reviews have provided broad overviews of AI in healthcare, they have often lacked a targeted focus on AI-based interventions evaluated through randomised controlled trials (RCTs).

To address this gap, a systematic review was conducted to identify and characterise diagnostic and therapeutic interventions based on AI, as evaluated in RCTs.

This review offers a structured synthesis of current applications and reported outcomes compared to conventional methods, highlighting areas of clinical promise, heterogeneity across medical fields, and directions for future research and safe implementation [23].

II. Methodology

A. Design and Eligibility Criteria

This systematic review followed the PRISMA 2020 guidelines to ensure comprehensive and transparent reporting [24]. The PRISMA 2020 checklist is included as Supplementary Material 1. The objective of this review was to identify and synthesise randomised controlled trials (RCTs) that evaluated diagnostic or therapeutic interventions based on artificial intelligence (AI), compared to conventional approaches. The research question was structured using the PICO framework as follows:

- Population (P): Patients with specific medical conditions receiving AI-based diagnostic or therapeutic interventions.
- Intervention (I): AI-based diagnostic or therapeutic interventions.
- Comparison (C): Conventional diagnostic or therapeutic methods, placebo, or standard care.
- Outcome (O): Diagnostic accuracy, treatment efficacy, precision, and/or side effects.

1) Inclusion criteria were: (i) RCTs evaluating diagnostic or therapeutic interventions supported by AI; (ii) studies including patients of any age, sex, or clinical condition; (iii) trials comparing AI-based approaches with conventional diagnostic or treatment methods, placebo, or standard care; and (iv) studies reporting quantitative clinical outcomes.

2) Exclusion criteria were: (i) non-randomised studies (e.g., observational, cross-sectional, cohort, or case-control designs); (ii) RCTs without a comparator group using conventional methods; (iii) studies not reporting relevant clinical outcomes; (iv) duplicate publications; (v) editorials, conference abstracts, letters, reviews, or protocols.

The review protocol was prospectively registered in the International Prospective Register of Systematic Reviews (PROSPERO) under registration number CRD42024583396, on 26 August 2024.

B. Information Sources and Search Strategy

A systematic search was conducted across PubMed, Scopus, and Web of Science for studies published between January 2015 and June 2024. The search strategy combined controlled vocabulary and free-text terms related to artificial intelligence and clinical interventions. Core search terms included “artificial intelligence”, “machine learning”, “deep learning”, “neural networks”, “natural language processing” combined with clinical concepts such as “diagnosis”, “treatment”, “therapeutic”, “patient care”, and study design filters (“randomised controlled trial”, “clinical trial”).

Specific exclusion terms were also applied to remove non-relevant designs, such as reviews, meta-analyses, case reports, editorials, and observational studies. Only original articles classified as clinical trials were retained. No restrictions were applied regarding patient population, but searches were limited to articles published in English or Spanish.

The detailed search strategies for each database, including Boolean operators, limits, and filters, are provided in Supplementary Material 2. A total of 861 records were retrieved from PubMed, 1,359 from Scopus, and 306 from Web of Science.

C. Selection and Data Collection Process

All identified references were imported into Rayyan [25, 26]. where duplicates were automatically removed. Two reviewers (SPC, OMS) independently screened titles and abstracts, followed by full-text review of potentially eligible articles. Any disagreements were resolved through discussion with a third reviewer (RFP).

From the 2,526 records initially retrieved, 2,096 remained after duplicate removal. After screening 137 titles and abstracts, 91 full-text articles were assessed for eligibility. Of these, 73 were excluded for not meeting the inclusion criteria (detailed in Supplementary Material 3). Ultimately, 13 RCTs were included in the final analysis (Fig. 1).

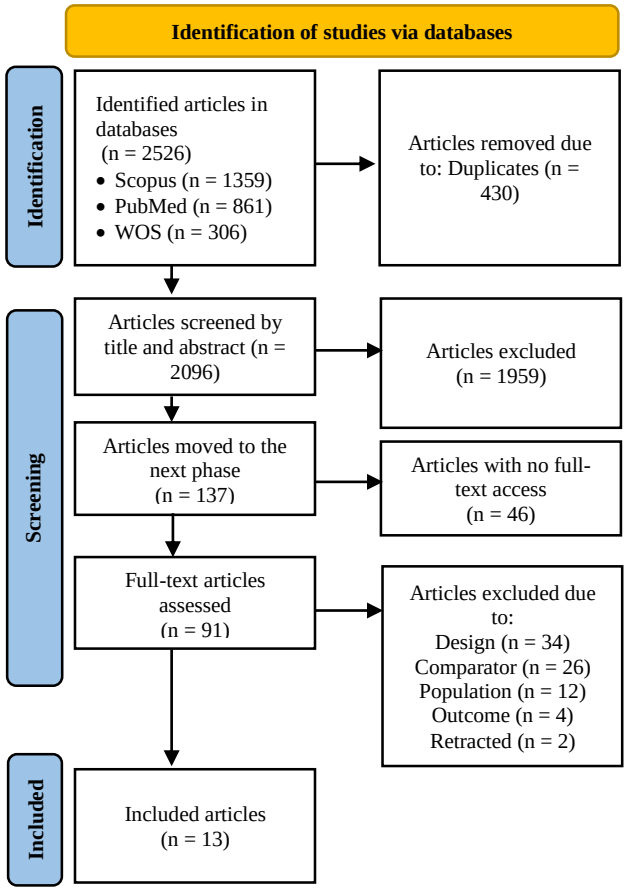


Fig. 1. Flow diagram of studies inclusion.

Data extraction was conducted independently by two reviewers using a pre-piloted form. Discrepancies were resolved by consensus.

D. Variables Collected

The primary outcomes of interest were diagnostic accuracy, treatment efficacy, and precision. Secondary outcomes included side effects and other clinically relevant metrics reported by the included studies. Each outcome was defined operationally according to the measures used in the original trials.

E. Risk of Bias Assessment

The risk of bias of the included RCTs was independently assessed by two reviewers (SPC, RFP) using the Cochrane RoB1 tool [27]. This tool evaluates six domains: random sequence generation, allocation concealment, blinding of participants, personnel and outcome assessors, incomplete outcome data, selective reporting, and other potential sources of bias. Disagreements were resolved with input from a third reviewer (OJF).

The overall risk of bias for each trial was categorised as low, unclear, or high, based on the assessment across domains.

III. Results

A. Study Selection

The breakdown of results by database followed the search strategy based on the following string: "(artificial intelligence" OR "AI" OR "machine learning" OR "deep learning" OR "neural network*" OR "natural language processing") AND ("medicine" OR "medical" OR "healthcare" OR "clinical" OR "health" OR "patient care" OR "diagnosis" OR "treatment" OR "biomedicine")". A total of 2,526 open-access records were identified, considering documents in English and Spanish from the years 2015 to 2024 for Scopus, WoS, and PubMed.

After removing duplicates, 2,096 records were screened by title and abstract. Following this initial screening, 137 articles were deemed potentially relevant and selected for full-text review. However, full text was not accessible for 46 of these articles. The remaining 91 articles underwent thorough full-text evaluation, leading to the exclusion of 73 studies due to various reasons, such as not meeting the inclusion criteria or lacking sufficient data for assessment. Ultimately, 13 randomised controlled trials (RCTs) with a combined total of 10,566 participants were included in the systematic review (Fig. 1).

B. Risk of Bias

To enhance the rigor of the study, RoB 1 (Cochrane Tool) was employed for RCT. This comprehensive analysis covered various types of bias that could compromise the internal validity of the RCTs included in this review, including random sequence generation and allocation concealment to identify potential selection biases. Additionally, participant, personnel, and assessor blinding were examined to address performance and detection biases. Incomplete outcome data and selective reporting were also reviewed, as they might reflect attrition and reporting biases, respectively. Considering these factors, along with other potential biases, resulted in a thorough evaluation aimed at improving the study's quality and its conclusions, as detailed in Fig. 2. Table I shows characteristics of included randomized controlled trials.

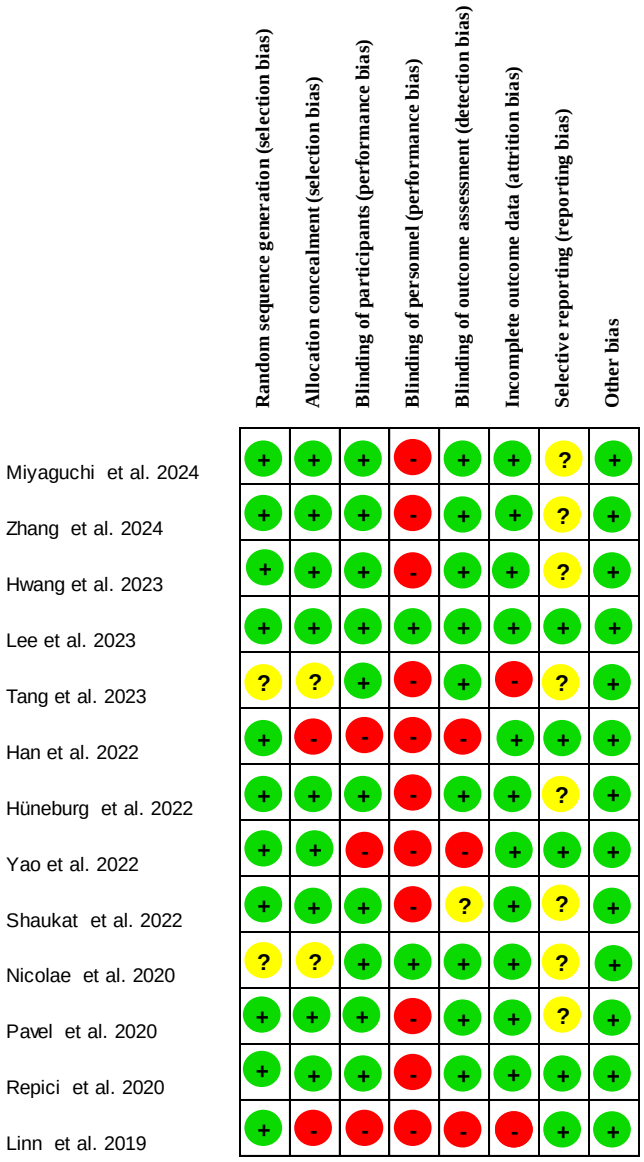


Fig. 2. Risk of bias summary across included studies. Note. Green represents low risk of bias; yellow represents unclear risk of bias; red represents high risk of bias.

TABLE I. CHARACTERISTICS OF INCLUDED RANDOMIZED CONTROLLED TRIALS

Author (year)	Medical Area	Population / Sample size	Intervention	Intervention Type	Comparator	AI Technology Used	Main Conclusion
Gastroenterology							
Miyaguchi (2024) [28]	Gastroenterology	Adults ≥20 undergoing colonoscopy / 800 (I:400, C:400)	Linked-color imaging colonoscopy with AI assistance	Diagnostic	Linked-color imaging colonoscopy without AI	CADEYE (Fujifilm EW10-EC02)	AI-assisted colonoscopy improved adenoma detection rate (ADR).
Zhang (2024) [29]	Gastroenterology	Adults 18–75 requiring colonoscopy / 1360 (I:680, C:680)	White-light colonoscopy with CADe	Diagnostic	White-light colonoscopy without CADe	EndoAngel (Wuhan ENDOANGEL)	Improved sensitivity and reduced polyp miss rate.
Tang (2023) [32]	Gastroenterology	Adults with pancreatic masses / 44 (I:21, C:23)	CH-EUS MASTER-guided EUS-FNA	Diagnostic	Conventional EUS-FNA	CH-EUS MASTER (AI-guided contrast-enhanced EUS)	Improved diagnostic accuracy and yield.
Hüneburg (2022) [34]	Gastroenterology	Adults ≥18 with Lynch syndrome / 96 (I:50, C:46)	AI-assisted colonoscopy	Diagnostic	High-definition white-light endoscopy (HD-WLE)	CAD-EYE (Fujifilm EW10-EC02)	Improved flat adenoma detection; overall ADR similar.
Yao (2022) [35]	Gastroenterology	Adults >18 undergoing colonoscopy / 1076 (I:805, C:271)	CADe, CAQ, and combined systems	Diagnostic	Colonoscopy without AI	CADe (YOLO V3), CAQ (AI quality control)	Combined AI improved ADR; CAQ was more effective than CADe alone.
Shaukat (2022) [36]	Gastroenterology	Adults ≥40 / 1369 (I:682, C:677)	Colonoscopy with CADe	Diagnostic	Standard colonoscopy	CADe (Computer-Aided Detection)	Increased ADR and adenomas per colonoscopy (APC).
Repici (2020) [39]	Gastroenterology	Adults 40–80 undergoing colonoscopy / 685 (I:341, C:340)	Colonoscopy with CADe	Diagnostic	Colonoscopy without CADe	GI-Genius (Medtronic)	Increased ADR without prolonging procedure time.
Radiology							
Hwang (2023) [30]	Radiology	Adults ≥19 with respiratory symptoms / 3576 (I:1761, C:1815)	Chest radiograph interpretation with AI-CAD	Diagnostic	Standard chest radiograph interpretation	Lunit INSIGHT CXR v2.0.2.0	No improvement in sensitivity or false positives.
Lee (2023) [31]	Radiology	Adults ≥20 undergoing chest X-ray / 329 (I:166, C:163)	AI-assisted interpretation of thoracic abnormalities	Diagnostic	Standard interpretation without AI	Lunit INSIGHT CXR v2.0.3.0	Improved non-radiologist diagnostic accuracy.
Dermatology							
Han (2022) [33]	Dermatology	Adults >19 with suspicious lesions / 576 (I:305, C:292)	Real-time AI-assisted diagnosis	Diagnostic	Diagnosis without AI	Model Dermatology (Multiclass AI)	Significantly improved accuracy, especially in non-experts.
Oncology							
Nicolae (2020) [37]	Oncology	Prostate cancer patients / 41 (I:21, C:20)	Machine learning-based treatment planning	Therapeutic	Manual planning	PIPA (Prostate Implant Planning Algorithm)	Comparable dosimetry, with time and efficiency gains.
Neurology							
Pavel (2020) [38]	Neurology	Neonates (36–44 weeks) / 264 (I:132, C:132)	EEG with ANSeR for seizure detection	Diagnostic	Conventional EEG monitoring	ANSeR (Algorithm for Neonatal Seizure Recognition)	Better seizure-hour identification; no difference at patient level.
Ophthalmology							
Linn (2019) [40]	Ophthalmology	Pediatric patients ≤14 years / 350 (I:350, C:350)	Diagnosis using CC-Cruiser	Diagnostic	Diagnosis by senior consultants	CC-Cruiser (Ophthalmic AI platform)	Lower diagnostic accuracy, but faster with higher satisfaction.

Note. I: Intervention group; C: Control group; AI: Artificial Intelligence; CADe: Computer-Aided Detection; CAQ: Computer-Aided Quality Improvement; CAD: Computer-Aided Diagnosis; ADR: Adenoma Detection Rate; ED: Emergency Department; CR: Chest Radiography; CXR: Chest X-ray; EUS-FNA: Endoscopic Ultrasound-Guided Fine-Needle Aspiration; CH-EUS MASTER: AI-assisted Contrast-Enhanced Harmonic Endoscopic Ultrasound system; HD-WLE: High-Definition White-Light Endoscopy; PIPA: Prostate Implant Planning Algorithm; ANSeR: Algorithm for Neonatal Seizure Recognition; EEG: Electroencephalogram.

IV. DISCUSSION

This systematic review synthesised the current landscape of diagnostic and therapeutic interventions based on artificial intelligence (AI) as evaluated in randomised controlled trials (RCTs). The included studies span multiple medical specialties, offering a structured overview of how AI technologies are being implemented in clinical settings and what outcomes have been reported in comparison with conventional approaches.

In gastroenterology, the integration of AI into endoscopic practice has consistently demonstrated clinically meaningful improvements. Multiple studies [28] and [39] confirmed that AI-assisted colonoscopy significantly increased adenoma detection rates (ADR), a well-established surrogate marker for colorectal cancer prevention. Complementarily, the use of Computer-Aided Detection (CADe) systems [29], [36] enhanced sensitivity for polyp detection without increasing unnecessary resections of non-neoplastic lesions, reinforcing their safety profile [22]. Importantly, AI demonstrated added value in specific high-risk populations: study [34] reported superior detection of flat adenomas in patients with Lynch syndrome, which are frequently overlooked by conventional methods. Beyond colonoscopy, AI has also optimized diagnostic yield in other domains, as evidenced by the CH-EUS MASTER system [32], which improved accuracy during EUS-FNA of pancreatic masses. Furthermore, the combined use of AI-driven detection and quality control systems [35] suggested a synergistic effect on real-time ADR, although this benefit was not replicated when each system was applied independently. Collectively, these findings not only highlight AI's capacity to enhance early lesion detection but also illustrate its potential to reshape preventive strategies in gastrointestinal oncology [41]. By enabling real-time visual analysis and identification of subtle mucosal changes often imperceptible to human observers, gastroenterology emerges as one of the most advanced fields for clinical AI implementation [42].

In radiology, the findings were more heterogeneous. Study [30] reported no significant improvement in sensitivity or false-positive rates when chest radiographs were interpreted with AI assistance for the diagnosis of acute thoracic diseases. Conversely, study [31] demonstrated that an AI-assisted diagnostic protocol based on deep learning significantly enhanced the performance of non-radiologist physicians in detecting pulmonary abnormalities. These discrepant results may be explained by variations in the maturity of the algorithms, the expertise of end-users, and the clinical contexts in which the tools were applied. Collectively, they highlight the pressing need for validation studies that account for both technological development and real-world implementation challenges.

In dermatology, study [33] showed that real-time AI assistance substantially improved diagnostic accuracy among non-expert physicians evaluating suspicious skin lesions. Interestingly, the greatest benefit was seen among trainees with limited dermatological experience, suggesting that AI functions not merely as a diagnostic adjunct but also as a pedagogical tool. This points to an underexplored opportunity for AI to strengthen medical education by supporting clinical reasoning and decision-making in early-career professionals.

In oncology, study [37] showed that machine learning-based treatment planning for prostate cancer, specifically using the Prostate Implant Planning Algorithm (PIPA), improved time efficiency and streamlined workflow without compromising dosimetric quality compared to manual planning. These results highlight AI's potential to optimize operational processes while maintaining treatment precision.

In neurology, study [38] assessed the use of the ANSeR algorithm for neonatal seizure recognition. Although AI improved the identification of seizure hours, it did not enhance detection at the patient level when compared with conventional EEG. This suggests that, while AI can assist in data interpretation, its current clinical contribution to neonatal seizure diagnostics remains limited.

In ophthalmology, study [40] evaluated the CC-Cruiser system and found that, although it facilitated faster diagnostic processes and improved patient satisfaction, its diagnostic accuracy and therapeutic decision-making were inferior to those of senior consultants. These results underscore the potential of AI to enhance efficiency and patient experience, but they also reveal critical limitations in replicating expert-level judgement. This suggests that, at present, AI should be regarded as an adjunctive tool rather than a replacement in complex clinical decision-making, highlighting the need for further refinement and validation before broader clinical integration.

Overall, the included studies demonstrate a broad spectrum of AI applications across medical disciplines. However, the representation across specialties is uneven**,** with gastroenterology dominating the evidence base, while areas such as neurology [38] and ophthalmology [40] remain underrepresented. This limits the generalizability of findings to less-studied fields and highlights the need for more diverse and comprehensive RCTs in AI research. Additionally, the majority of included studies were conducted in technologically advanced, well-resourced hospital settings. Consequently, their applicability to low-resource or primary care environments remains uncertain.

The certainty of evidence also varied. In gastroenterology, robust findings from multiple high-powered RCTs (e.g., [28], [29], [39]) provide strong support for the use of AI. In contrast, the mixed results in radiology [30], [31] and limited data in oncology and neurology reduce confidence in the broader applicability of these technologies. The variability in AI system types, clinical indications, and user expertise further complicates the ability to draw uniform conclusions across studies.

Despite its strengths, this systematic review has several limitations. First, although the search strategy was extensive, it is possible that relevant studies were missed due to language restrictions or the stringent inclusion criteria. Second, heterogeneity across AI systems, patient populations, and outcome measures reduced the comparability of results and precluded meta-analysis. However, this same heterogeneity reflects the diversity and innovation within the field, illustrating the breadth of AI's applicability to various clinical challenges.

Notably, this review adheres to rigorous methodological standards, including AMSTAR-2 compliance, a predefined protocol, and a structured risk of bias assessment, which

collectively reinforce the validity and transparency of the findings. By restricting the review to randomised controlled trials, the study focused on high-quality evidence, ensuring that the conclusions are grounded in rigorous experimental data.

In addition to evaluating clinical performance, this review indirectly reflects the growing sophistication and practical deployment of machine learning algorithms in healthcare settings. Most of the AI-based interventions included in the analysed RCTs involve deep learning models, computer vision tools, or natural language processing systems integrated into real-time clinical workflows. These findings not only demonstrate the clinical utility of AI but also highlight the importance of interdisciplinary collaboration between clinicians, data scientists, and engineers. Future studies should document more precisely the algorithmic architecture, training datasets, performance metrics, and update protocols of the AI systems being evaluated, to ensure reproducibility and transparency across both medical and computational communities.

Collectively, these findings illustrate both the promise and the fragility of the current evidence base. Gastroenterology emerges as a model of successful integration of AI into routine practice, whereas other specialties remain in earlier stages of exploration, with evidence that is either inconsistent or limited in scope. This uneven development highlights the urgent need for methodological standardisation, greater representation of diverse healthcare contexts, and longitudinal validation studies to assess sustainability of outcomes over time. Addressing these gaps will not only advance the safe and equitable deployment of AI across medical disciplines but also ensure that its benefits extend beyond technologically advanced centres to the broader spectrum of global healthcare.

Future research should prioritise underexplored medical specialties, assess AI interventions across diverse healthcare settings—including low-resource and primary care environments—and examine the impact of user training and experience on clinical outcomes. Longitudinal and multicentre studies are encouraged to evaluate the sustainability and generalisability of AI-assisted interventions. Moreover, developing standardised methodological frameworks and reporting guidelines will facilitate comparability, reproducibility, and evidence-informed adoption of AI technologies across clinical disciplines.

V. Conclusion

This systematic review identified and synthesised randomised controlled trials evaluating AI-based diagnostic and therapeutic interventions compared to conventional methods. The findings highlight that AI technologies can improve diagnostic accuracy and clinical performance, particularly in fields such as gastroenterology and dermatology. However, the variability of outcomes observed in specialties such as radiology, neurology and ophthalmology suggest that the clinical utility of AI is highly context dependent and influenced by the nature of the task, technological design and user expertise.

These results underscore the importance of a critical and context-sensitive approach to the integration of AI in healthcare. While promising results support its implementation in specific domains, the observed heterogeneity indicates that AI is not a universal solution. A deeper understanding of the conditions under which AI adds the most value is essential, considering clinical context, provider training, technological maturity and patient characteristics.

To advance responsible and effective integration of AI into clinical practice, future research should expand evaluation across diverse healthcare settings and populations. The development of standardised protocols and implementation guidelines is needed to ensure consistent and evidence-informed use.

Additionally, studies should examine the impact of user training and experience on AI outcomes, and ensure that the deployment of these technologies promotes equity in access and benefit across all socioeconomic and geographic contexts. In summary, this systematic review provides an updated and evidence-based overview of AI applications in randomised controlled trials, offering a critical foundation to guide future research and the responsible integration of AI into clinical practice.

References

[1] Silcox, C., Zimlichmann, E., Huber, K., Rowen, N., Saunders, R., McClellan, M., Kahn, C.N., Salzberg, C.A., Bates, D.W.: The potential for artificial intelligence to transform healthcare: perspectives from international health leaders. npj Digit. Med. 7, 88, 2024. https://doi.org/10.1038/s41746-024-01097-6.

[2] Da Silva, R.G.L.: The advancement of artificial intelligence in biomedical research and health innovation: challenges and opportunities in emerging economies. Global Health. 20, 44, 2024. https://doi.org/10.1186/s12992-024-01049-5.

[3] Gao, S., Xu, Z., Kang, W., Lv, X., Chu, N., Xu, S., Hou, D.: Artificial intelligence-driven computer aided diagnosis system provides similar diagnosis value compared with doctors' evaluation in lung cancer screening. BMC Med Imaging. 24, 141, 2024. https://doi.org/10.1186/s12880-024-01288-3.

[4] Knudsen, J.E., Ghaffar, U., Ma, R., Hung, A.J.: Clinical applications of artificial intelligence in robotic surgery. J Robotic Surg. 18, 102, 2024. https://doi.org/10.1007/s11701-024-01867-0.

[5] Bowman, M.A., Seehusen, D.A., Britz, J., Ledford, C.J.W.: Research to Improve Clinical Care in Family Medicine: Big Data, Telehealth, Artificial Intelligence, and More. J Am Board Fam Med. 37, 161–164, 2024. https://doi.org/10.3122/jabfm.2024.240050R0.

[6] Green, M.: Could artificial intelligence improve patient care and physician workload? Can Fam Physician. 70, 216–216, 2024. https://doi.org/10.46747/cfp.7003216.

[7] Zhang, L., Wong, C., Li, Y., Huang, T., Wang, J., Lin, C.: Artificial intelligence assisted diagnosis of early tc markers and its application. Discov Onc. 15, 172, 2024. https://doi.org/10.1007/s12672-024-01017-w.

[8] Chen, A., Wang, C., Zhang, X.: Reflection on the equitable attribution of responsibility for artificial intelligence-assisted diagnosis and treatment decisions. Intelligent Medicine. 3, 139–143, 2023. https://doi.org/10.1016/j.imed.2022.04.002.

[9] Cohen, R.Y., Kovacheva, V.P.: A Methodology for a Scalable, Collaborative, and Resource-Efficient Platform, MERLIN, to Facilitate Healthcare AI Research. IEEE J. Biomed. Health Inform. 27, 3014–3025, 2023. https://doi.org/10.1109/JBHI.2023.3259395.

[10] A, A., Dahan, F., Alroobaea, R., Alghamdi, Wael.Y., Mustafa Khaja Mohammed, Hajjej, F., Deema Mohammed Alsekait, Raahemifar, K.: A smart IoMT based architecture for E-healthcare patient monitoring system using artificial intelligence algorithms. Front. Physiol. 14, 1125952, 2023. https://doi.org/10.3389/fphys.2023.1125952.

[11] Akyüz, K., Cano Abadía, M., Goisauf, M., Mayrhofer, M.Th.: Unlocking the potential of big data and AI in medicine: insights from biobanking.

Front. Med. 11, 1336588, 2024. https://doi.org/10.3389/fmed.2024.1336588.

[12] Witkowski, K., Okhai, R., Neely, S.R.: Public perceptions of artificial intelligence in healthcare: ethical concerns and opportunities for patient-centered care. BMC Med Ethics. 25, 74, 2024. https://doi.org/10.1186/s12910-024-01066-4.

[13] Palmer, A., Schwan, D.: More Process, Less Principles: The Ethics of Deploying AI and Robotics in Medicine. Camb Q Healthc Ethics. 33, 121–134, 2024. https://doi.org/10.1017/S0963180123000087.

[14] Huang, S.-H., Peng, B.-R., Lin, C.-S., Tsai, H.-C., Pan, L.-F., Pan, L.-K.: Potential risk quantification from multiple biological factors via the inverse problem algorithm as an artificial intelligence tool in clinical diagnosis. THC. 31, 69–79, 2023. https://doi.org/10.3233/THC-236008.

[15] Aquino, Y.S.J., Carter, S.M., Houssami, N., Braunack-Mayer, A., Win, K.T., Degeling, C., Wang, L., Rogers, W.A.: Practical, epistemic and normative implications of algorithmic bias in healthcare artificial intelligence: a qualitative study of multidisciplinary expert perspectives. J Med Ethics. jme-2022-108850, 2023. https://doi.org/10.1136/jme-2022-108850.

[16] Yoo, J., Hur, S., Hwang, W., Cha, W.C.: Healthcare Professionals' Expectations of Medical Artificial Intelligence and Strategies for its Clinical Implementation: A Qualitative Study. Healthc Inform Res. 29, 64–74, 2023. https://doi.org/10.4258/hir.2023.29.1.64.

[17] Eskandarani, R., Almuhainy, A., Alzahrani, A.: Creating a master training rotation schedule for emergency medicine residents and challenges in using artificial intelligence. Int J Emerg Med. 17, 84, 2024. https://doi.org/10.1186/s12245-024-00657-7.

[18] Krajcer, Z.: Artificial Intelligence for Education, Proctoring, and Credentialing in Cardiovascular Medicine. Texas Heart Institute Journal. 49, e217572, 2022. https://doi.org/10.14503/THIJ-21-7572.

[19] Bekbolatova, M., Mayer, J., Ong, C.W., Toma, M.: Transformative Potential of AI in Healthcare: Definitions, Applications, and Navigating the Ethical Landscape and Public Perspectives. Healthcare. 12, 125, 2024. https://doi.org/10.3390/healthcare12020125.

[20] Oniani, D., Hilsman, J., Peng, Y., Poropatich, R.K., Pamplin, J.C., Legault, G.L., Wang, Y.: Adopting and expanding ethical principles for generative artificial intelligence from military to healthcare. npj Digit. Med. 6, 225, 2023. https://doi.org/10.1038/s41746-023-00965-x.

[21] Ranjbar, A., Mork, E., Ravn, J., Brøgger, H., Myrseth, P., Østrem, H.P., Hallock, H.: Managing Risk and Quality of AI in Healthcare: Are Hospitals Ready for Implementation? RMHP. Volume 17, 877–882, 2024. https://doi.org/10.2147/RMHP.S452337.

[22] Macrae, C.: Managing risk and resilience in autonomous and intelligent systems: Exploring safety in the development, deployment, and use of artificial intelligence in healthcare. Risk Analysis. risa.14273 ,2024. https://doi.org/10.1111/risa.14273.

[23] Alami, H., Lehoux, P., Papoutsi, C., Shaw, S.E., Fleet, R., Fortin, J.-P.: Understanding the integration of artificial intelligence in healthcare organisations and systems through the NASSS framework: a qualitative study in a leading Canadian academic centre. BMC Health Serv Res. 24, 701, 2024. https://doi.org/10.1186/s12913-024-11112-x.

[24] Page, M.J., McKenzie, J.E., Bossuyt, P.M., Boutron, I., Hoffmann, T.C., Mulrow, C.D., Shamseer, L., Tetzlaff, J.M., Akl, E.A., Brennan, S.E., Chou, R., Glanville, J., Grimshaw, J.M., Hróbjartsson, A., Lalu, M.M., Li, T., Loder, E.W., Mayo-Wilson, E., McDonald, S., McGuinness, L.A., Stewart, L.A., Thomas, J., Tricco, A.C., Welch, V.A., Whiting, P., Moher, D.: The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. BMJ. n71 ,2021. https://doi.org/10.1136/bmj.n71.

[25] Valizadeh, A., Moassefi, M., Nakhostin-Ansari, A., Hosseini Asl, S.H., Saghab Torbati, M., Aghajani, R., Maleki Ghorbani, Z., Faghani, S.: Abstract screening using the automated tool Rayyan: results of effectiveness in three diagnostic test accuracy systematic reviews. BMC Med Res Methodol. 22, 160, 2022. https://doi.org/10.1186/s12874-022-01631-8.

[26] Yu, F., Liu, C., Sharmin, S.: Performance, Usability, and User Experience of Rayyan for Systematic Reviews. Proceedings of the Association for Information Science and Technology. 59, 843–844, 2022. https://doi.org/10.1002/pra2.745.

[27] Higgins, J.P., Savović, J., Page, M.J., Elbers, R.G., Sterne, J.A.: Assessing risk of bias in a randomized trial. In: Higgins, J.P.T., Thomas, J., Chandler, J., Cumpston, M., Li, T., Page, M.J., and Welch, V.A. (eds.) Cochrane Handbook for Systematic Reviews of Interventions. pp. 205–228. Wiley .2019. https://doi.org/10.1002/9781119536604.ch8.

[28] Miyaguchi, K., Tsuzuki, Y., Hirooka, N., Matsumoto, H., Ohgo, H., Nakamoto, H., Imaeda, H.: Linked-color imaging with or without artificial intelligence for adenoma detection: a randomized trial. Endoscopy. 56, 376–383, 2024. https://doi.org/10.1055/a-2239-8145.

[29] Zhang, H., Wu, Q., Sun, J., Wang, J., Zhou, L., Cai, W., Zou, D.: A computer-aided system improves the performance of endoscopists in detecting colorectal polyps: a multi-center, randomized controlled trial. Front. Med. 10, 1341259, 2024. https://doi.org/10.3389/fmed.2023.1341259.

[30] Hwang, E.J., Goo, J.M., Nam, J.G., Park, C.M., Hong, K.J., Kim, K.H.: Conventional Versus Artificial Intelligence-Assisted Interpretation of Chest Radiographs in Patients With Acute Respiratory Symptoms in Emergency Department: A Pragmatic Randomized Clinical Trial. Korean J Radiol. 24, 259, 2023. https://doi.org/10.3348/kjr.2022.0651.

[31] Lee, H.W., Jin, K.N., Oh, S., Kang, S.-Y., Lee, S.M., Jeong, I.B., Son, J.W., Han, J.H., Heo, E.Y., Lee, J.G., Kim, Y.J., Kim, E.Y., Cho, Y.J.: Artificial Intelligence Solution for Chest Radiographs in Respiratory Outpatient Clinics: Multicenter Prospective Randomized Clinical Trial. Annals ATS. 20, 660–667, 2023. https://doi.org/10.1513/AnnalsATS.202206-481OC.

[32] Tang, A., Tian, L., Gao, K., Liu, R., Hu, S., Liu, J., Xu, J., Fu, T., Zhang, Z., Wang, W., Zeng, L., Qu, W., Dai, Y., Hou, R., Tang, S., Wang, X.: Contrast - enhanced harmonic endoscopic ultrasound (CH - EUS) MASTER: A novel deep learning - based system in pancreatic mass diagnosis. Cancer Medicine. 12, 7962–7973, 2023. https://doi.org/10.1002/cam4.5578.

[33] Han, S.S., Kim, Y.J., Moon, I.J., Jung, J.M., Lee, M.Y., Lee, W.J., Won, C.H., Lee, M.W., Kim, S.H., Navarrete-Dechent, C., Chang, S.E.: Evaluation of Artificial Intelligence–Assisted Diagnosis of Skin Neoplasms: A Single-Center, Paralleled, Unmasked, Randomized Controlled Trial. Journal of Investigative Dermatology. 142, 2353-2362.e2. 2022. https://doi.org/10.1016/j.jid.2022.02.003.

[34] Hüneburg, R., Bucksch, K., Schmeißer, F., Heling, D., Marwitz, T., Aretz, S., Kaczmarek, D.J., Kristiansen, G., Hommerding, O., Strassburg, C.P., Engel, C., Nattermann, J.: Real - time use of artificial intelligence (CADEYE) in colorectal cancer surveillance of patients with Lynch syndrome—A randomized controlled pilot trial (CADLY). UEG Journal. 11, 60–68, 2023. https://doi.org/10.1002/ueg2.12354.

[35] Yao, L., Zhang, L., Liu, J., Zhou, W., He, C., Zhang, J., Wu, L., Wang, H., Xu, Y., Gong, D., Xu, M., Li, X., Bai, Y., Gong, R., Sharma, P., Yu, H.: Effect of an artificial intelligence-based quality improvement system on efficacy of a computer-aided detection system in colonoscopy: a four-group parallel study. Endoscopy. 54, 757–768, 2022. https://doi.org/10.1055/a-1706-6174.

[36] Shaukat, A., Lichtenstein, D.R., Somers, S.C., Chung, D.C., Perdue, D.G., Gopal, M., Colucci, D.R., Phillips, S.A., Marka, N.A., Church, T.R., Brugge, W.R., Thompson, R., Chehade, R., Loew, B., Downing, J., Vermillion, J., Borges, L., Rajbhandari, R., Schafer, T., Coban, S., Richter, J., Carolan, P., Colizzo, F., Jeong, T., DelSignore, M., Asher, S., McCabe, R., Van Handel, D., Cinnor, B., Mitlyng, B., Sherman, C., Feldshon, S.D., Lounsbury, A., Thompson, A., Duggirala, A., Davies, I., Huang, C., Bliss, C., Mohanty, A., Sina, O., Mendez, J., Iwan, A., Stromberg, J., Ng, J., Erisson, L., Golland, P., Wang, D., Wlodkowski, E., Carlin, J., Javia, P., Chavali, N., Wang, A., Little, J., Hunsberger, C.: Computer-Aided Detection Improves Adenomas per Colonoscopy for Screening and Surveillance Colonoscopy: A Randomized Trial. Gastroenterology. 163, 732–741, 2022. https://doi.org/10.1053/j.gastro.2022.05.028.

[37] Nicolae, A., Semple, M., Lu, L., Smith, M., Chung, H., Loblaw, A., Morton, G., Mendez, L.C., Tseng, C.-L., Davidson, M., Ravi, A.: Conventional vs machine learning–based treatment planning in prostate brachytherapy: Results of a Phase I randomized controlled trial. Brachytherapy. 19, 470–476, 2020. https://doi.org/10.1016/j.brachy.2020.03.004.

[38] Pavel, A.M., Rennie, J.M., De Vries, L.S., Blennow, M., Foran, A., Shah, D.K., Pressler, R.M., Kapellou, O., Dempsey, E.M., Mathieson, S.R., Pavlidis, E., Van Huffelen, A.C., Livingstone, V., Toet, M.C., Weeke, L.C., Finder, M., Mitra, S., Murray, D.M., Marnane, W.P., Boylan, G.B.: A machine-learning algorithm for neonatal seizure recognition: a multicentre, randomised, controlled trial. The Lancet Child & Adolescent Health. 4, 740–749, 2020. https://doi.org/10.1016/S2352-4642(20)30239-X.

[39] Repici, A., Badalamenti, M., Maselli, R., Correale, L., Radaelli, F., Rondonotti, E., Ferrara, E., Spadaccini, M., Alkandari, A., Fugazza, A., Anderloni, A., Galtieri, P.A., Pellegatta, G., Carrara, S., Di Leo, M., Craviotto, V., Lamonaca, L., Lorenzetti, R., Andrealli, A., Antonelli, G., Wallace, M., Sharma, P., Rosch, T., Hassan, C.: Efficacy of Real-Time Computer-Aided Detection of Colorectal Neoplasia in a Randomized Trial. Gastroenterology. 159, 512-520.e7. 2020. https://doi.org/10.1053/j.gastro.2020.04.062.

[40] Lin, H., Li, R., Liu, Z., Chen, J., Yang, Y., Chen, H., Lin, Z., Lai, W., Long, E., Wu, X., Lin, D., Zhu, Y., Chen, C., Wu, D., Yu, T., Cao, Q., Li, X., Li, J., Li, W., Wang, J., Yang, M., Hu, H., Zhang, L., Yu, Y., Chen, X., Hu, J., Zhu, K., Jiang, S., Huang, Y., Tan, G., Huang, J., Lin, X., Zhang, X., Luo, L., Liu, Y., Liu, X., Cheng, B., Zheng, D., Wu, M., Chen, W., Liu, Y.: Diagnostic Efficacy and Therapeutic Decision-making Capacity of an Artificial Intelligence Platform for Childhood Cataracts in Eye Clinics: A Multicentre Randomized Controlled Trial. EClinicalMedicine. 9, 52–59, 2019. https://doi.org/10.1016/j.eclinm.2019.03.001.

[41] Uchikov, P., Khalid, U., Vankov, N., Kraeva, M., Kraev, K., Hristov, B., Sandeva, M., Dragusheva, S., Chakarov, D., Petrov, P., Dobreva-Yatseva, B., Novakov, I.: The Role of Artificial Intelligence in the Diagnosis and Treatment of Ulcerative Colitis. Diagnostics. 14, 1004, 2024. https://doi.org/10.3390/diagnostics14101004.

[42] Stafie, C.S., Sufaru, I.-G., Ghiciuc, C.M., Stafie, I.-I., Sufaru, E.-C., Solomon, S.M., Hancianu, M.: Exploring the Intersection of Artificial Intelligence and Clinical Healthcare: A Multidisciplinary Review. Diagnostics. 13, 1995, 2023. https://doi.org/10.3390/diagnostics13121995.

Transformer-Enabled Smartphone System for Intelligent Physical Activity Monitoring

Leping Zhang[1], Fengjiao Jiang[2]*, Guopeng Jia[3], Yue Wang[4]
Personnel Department, Shanghai Vocational College of Agriculture and Forestry, Shanghai, China[1]
Department of Intelligent Agricultural Engineering, Shanghai Vocational College of Agriculture and Forestry, Shanghai, China[2, 3, 4]

***Abstract*—This study addresses the prevalent decline in physical activity among university students in the contemporary information society, proposing an innovative deep learning-based framework for intelligent physical activity recognition. Central to this framework is the comprehensive utilization of high-precision Inertial Measurement Units (IMUs) integrated within smartphones, encompassing triaxial accelerometers, gyroscopes, and magnetometers, enabling multi-dimensional, real-time capture of students' daily activity postures. For algorithmic design, this research transcends traditional limitations by adopting the more advanced Transformer architecture as its core classifier. Through the distinct self-attention mechanism inherent to this architecture, the proposed method efficiently and precisely extracts critical spatiotemporal features from vast sensor data, thereby achieving accurate identification and classification of various physical activities, such as walking, running, and climbing stairs. Rigorous evaluation results demonstrate significant advantages in key performance metrics, including recognition accuracy, when compared to conventional recurrent neural networks (e.g., Long Short-Term Memory networks, Recurrent Neural Networks) and classic machine learning algorithms (e.g., Random Forest), with a validation accuracy reaching 93.97%. This forward-looking research outcome not only provides a reliable and efficient technological means for monitoring the physical activity status of university students but also establishes a robust data foundation for the future development and implementation of targeted health intervention measures.**

***Keywords*—Activity recognition; smartphone; transformer architecture; inertial measurement units**

I. Introduction

In recent years, the widespread adoption of smartphones and rapid advancements in mobile computing technology have positioned sensor-based Human Activity Recognition as a focal point in research fields such as health monitoring, intelligent assistance, and sports analytics [1]. Traditional methods for monitoring physical activity, such as questionnaires, field-based equipment assessments, or manual observation, suffer from limitations including subjectivity, high costs, and poor compliance, making them inadequate for large-scale, long-term, and precise monitoring. Consequently, there is an urgent need for intelligent, convenient, and efficient physical activity recognition technologies to address the shortcomings of conventional methods.

The perception of physical activities forms the cornerstone of intelligent recognition. Currently, the primary approaches to physical activity perception encompass time-series data-based sensing and image-based visual sensing [2]. The use of diverse sensor modalities for activity detection and classification has emerged as a transformative technology for real-time and autonomous monitoring in areas such as behavioral analysis in smart home environments, assisted living, daily activity monitoring, elderly care, rehabilitation, entertainment, and security surveillance. Wearable devices, smartphones [3], and ambient sensing equipment are equipped with an array of sensors, including accelerometers, gyroscopes, magnetometers, heart rate monitors, pressure sensors, and compact cameras, to facilitate activity recognition and monitoring [4]. These sensor data undergo preprocessing to extract feature sets, such as time-domain, frequency-domain, or wavelet transform features, which are then processed using machine learning algorithms to classify and continuously monitor human activities.

The perception of time-series data, such as acceleration signals, provides a convenient approach for activity recognition. By integrating these data with intelligent algorithms, accurate identification of human activities can be achieved. Yin et al. developed a two-stage method for detecting anomalous activities using wireless body sensors [5]. In the first stage, they trained a one-class Support Vector Machine (SVM) on common normal activities to filter out those highly likely to be normal. Subsequently, they used Kernel Nonlinear Regression (KNLR) to derive an anomaly activity model from the general normal model, reducing false positives in an unsupervised manner. Saeed et al. proposed a novel self-supervised technique for feature learning from sensor data, which did not require semantic labels (i.e., activity categories) [6]. They trained a multi-task temporal convolutional network to recognize transformations applied to input signals. By leveraging these transformations, they demonstrated that a simple binary auxiliary task could generate robust supervisory signals, extracting valuable features for downstream tasks. Janidarmian et al. performed an extensive analysis of feature representations and classification techniques for activity recognition, comparing 293 classifiers in the most comprehensive study to date [7]. They applied Principal Component Analysis (PCA) to reduce feature vector dimensionality while preserving essential information. Iloga et al. observed that deep learning techniques addressed some limitations but required significant computational resources and produced feature vectors with limited interpretability [8]. To address these challenges, they developed a Human Activity Recognition (HAR) technique based on Hidden Markov Models (HMMs).

*Corresponding Author.

Smartphones equipped with various sensors, such as accelerometers, gyroscopes, and magnetometers, can capture high-frequency data on posture, orientation, and velocity changes during human motion, generating continuous time-series data [9]. These data accurately capture unique patterns of periodic or non-periodic movements, such as running, jumping, and walking [10]. For instance, accelerometers record linear motion, while gyroscopes detect rotational motion, together providing a comprehensive representation of the body's dynamics in three-dimensional space [11]. Through preprocessing, feature extraction, and pattern recognition of sensor data, different physical activities can be effectively distinguished [12]. Compared to other wearable devices, smartphones, as indispensable tools in university students' daily lives, offer widespread availability and portability, making them an ideal platform for time-series data collection in physical activity sensing [13]. Lara et al. developed the Centinela system [14], which integrated acceleration data with vital signs to achieve high-accuracy activity recognition. The system identified five activities: walking, running, sitting, standing, and descending. They evaluated eight classifiers and three time window sizes, achieving an overall accuracy of 95.7%. Micucci et al. created a novel acceleration sample dataset [15], collected using an Android smartphone designed for human activity recognition and fall detection. The samples were categorized into 17 fine-grained classes and grouped into two coarse-grained classes: one comprising 9 types of Activities of Daily Living (ADL) and another including 8 types of falls. Mario proposed a novel mechanism for detecting specific activities using data from a single triaxial accelerometer [16]. They employed convolutional neural networks to automatically extract the most relevant features to characterize acceleration patterns, enabling cross-activity recognition.

With significant advancements in deep learning for image processing, intelligent recognition of human activities based on image data has seen rapid development. Niu et al. developed a framework for human activity detection and recognition in outdoor video surveillance applications [17]. They introduced an efficient activity representation method that identified distinct interaction patterns among groups based on simple statistical data from tracking trajectories, without requiring complex Markov chains, Hidden Markov Models (HMMs), or Coupled Hidden Markov Models (CHMMs). Sung et al. utilized RGBD sensors (Microsoft Kinect) as input devices and computed a set of features based on human posture, motion, and image and point cloud data [18]. They designed an algorithm based on a hierarchical Maximum Entropy Markov Model (MEMM), treating individual activities as compositions of sub-activities, and employed dynamic programming to infer a two-layer graph structure. Ni et al. proposed a novel framework for complex activity recognition and localization, effectively integrating information from grayscale and depth image channels across multiple layers of the video processing pipeline [19]. Koppula et al. modeled complex spatiotemporal relationships (termed affordances) between human postures and objects using Conditional Random Fields (CRFs), inferring multiple possible graph structures and approximating graphs with additive features for efficient dynamic programming [20]. Albanese et al. developed a computational framework for human activity representation based on Petri Nets [21]. They introduced the PPN-MPS algorithm for the first question and proposed two algorithms (naive PPN-MPA and PPN-MPA) for the second. Rodrigues et al. developed a multi-timescale model to capture temporal dynamics across different time scales, predicting future and past states for a given input posture trajectory [22].

In the domain of physical activity recognition, deep learning techniques have demonstrated exceptional capabilities in feature extraction and pattern recognition, establishing them as the most advanced and effective algorithmic paradigm. For time-series data collected by smartphones, Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and their variants, such as Long Short-Term Memory networks (LSTMs) and Gated Recurrent Units (GRUs), have been widely applied to physical activity recognition [23]. Additionally, models incorporating attention mechanisms further enhance the ability to capture critical motion features [24]. For image data, CNNs remain the dominant algorithm for physical activity recognition. Transfer learning based on pretrained ImageNet models significantly improves performance in image-based activity recognition tasks. By leveraging posture information derived from human skeleton keypoint detection, combined with algorithms like Graph Convolutional Networks (GCNs), complex physical activities can be analyzed more effectively through posture changes and action sequences. Furthermore, Transformer models have shown significant potential in processing image sequences and capturing global features, opening new research directions for image-based physical activity recognition [25].

Transformer technology continues to demonstrate significant potential in processing time-series and tabular data. To this end, this study proposes a novel method for intelligent recognition of physical activities in students using Transformer-based models and smartphone sensors. The method leverages high-precision inertial sensors embedded in smartphones, such as accelerometers and gyroscopes, to collect motion data. Through the Transformer neural network, feature extraction and classification are performed to automatically identify various physical activities, including walking, running, and climbing stairs. Compared to traditional time-series models such as Long Short-Term Memory (LSTM) networks, vanilla Recurrent Neural Networks (RNNs), and machine learning methods like Random Forest, the proposed method demonstrates a significant advantage in recognition accuracy. In terms of algorithmic architecture, this study pioneers the application of Transformer models in the domain of university students' daily activity recognition. Leveraging its distinctive self-attention mechanism, the proposed framework achieves efficient and precise extraction of crucial spatiotemporal features from massive sensor datasets. Regarding practical applications, the research not only develops a reliable and efficient monitoring system for collegiate physical activities but more significantly incorporates an innovative class-balancing mechanism that effectively mitigates recognition performance degradation caused by the data imbalance issue.

Section II provides a detailed introduction to the overall framework and the intelligent recognition algorithm of the proposed method. Section III conducts an in-depth analysis of

the utilized data, and Section IV evaluates the recognition performance of the proposed method in comparison with other approaches. Section V presents a discussion and analysis. Finally, Section VI summarizes the findings of this study.

II. Portable Perception and Intelligent Recognition of Human Activities

To accurately and comprehensively monitor and assess the types and states of physical activities among university students, this study innovatively proposes a portable monitoring method based on smartphone technology and deep learning. Its overall framework, clearly illustrated in Fig. 1, delineates the entire data flow and processing pipeline. This method capitalizes on the pervasive and ubiquitous nature of modern smartphones, specifically leveraging their integrated high-precision inertial sensors, such as triaxial accelerometers and triaxial gyroscopes. These sensors enable the portable sensing of human activities, capturing detailed motion data with remarkable accuracy, thereby establishing a robust foundation for subsequent analytical procedures. In the data processing phase, the raw sensor signals undergo a series of meticulous preprocessing steps. Specifically, the data is first subjected to precise segmentation in both the time and frequency domains, dividing it into fixed-length windows to ensure that each window encapsulates complete activity information. Subsequently, these segmented data are normalized to eliminate scale discrepancies between different sensors or individuals, ensuring data consistency. Following normalization, a rich set of time-domain and frequency-domain features is extracted from the signals, designed to capture the unique patterns of distinct physical activities. These extracted features collectively form a structured tabular dataset, where each sample comprises a carefully selected array of diverse features, preparing the data for input into the machine learning model.

Building upon this foundation, this research innovatively designs and proposes an optimized Transformer network model specifically tailored for this tabular data. This model integrates advanced deep learning techniques, including a unique mapping projection mechanism and a sophisticated positional encoding strategy. These techniques enable the Transformer network to effectively process the non-sequential structure inherent to tabular data and capture complex relationships and potential temporal dependencies among features (even though the data has been tabularized). Through this powerful model, we achieve highly accurate classification of various human activities, thereby robustly supporting the continuous monitoring and assessment of university students' activity types and states. This portable, efficient, and precise monitoring method promises to offer a revolutionary tool for managing the physical health of university students.

The physical activity recognition model proposed in this study leverages the robust modeling capabilities of the Transformer architecture combined with the characteristics of multidimensional feature data. Originally designed for natural language processing tasks, the Transformer model has gained widespread attention due to its superior sequence modeling and global information capture mechanisms. The model processes input tabular data through four core components—input projection, positional encoding, Transformer encoder, and classification head—to capture complex dependencies among features and generate probabilistic predictions for activity categories. Compared to traditional vanilla Recurrent Neural Networks (RNNs) or Convolutional Neural Networks (CNNs), the Transformer architecture employs a self-attention mechanism to process tabular data in parallel, significantly enhancing its ability to model multidimensional features while mitigating issues such as vanishing gradients. The input projection module maps raw tabular data features into a high-dimensional model space, improving feature representation. The positional encoding module introduces positional information to the tabular data, addressing the Transformer's inherent lack of sequence awareness. The Transformer encoder, utilizing multiple layers of multi-head self-attention mechanisms and feed-forward networks, captures intricate dependencies within the tabular data. The classification head pools the encoded sequence features and maps them to the category space, producing the final activity predictions. Each component incorporates recent advances in deep learning, ensuring the model's efficiency and robustness in handling time-series data. The following sections provide a detailed description of each module.

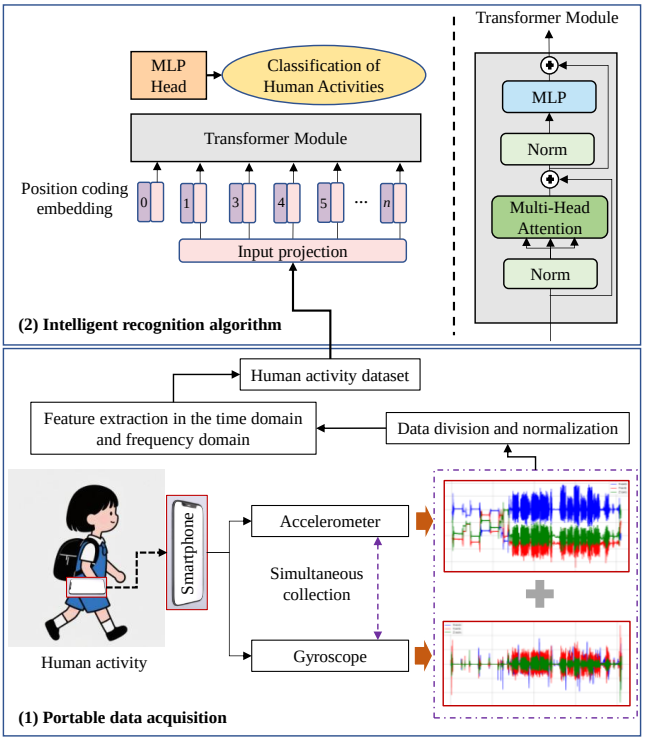


Fig. 1. Overall framework of the proposed method.

A. Projection of the Input Data

The input projection module aims to map raw tabular data features ($\mathbf{X} \in \mathrm{R}^{B \times D}$) to the model's internal dimension (d_{model}) to meet the input requirements of the Transformer encoder while enhancing feature representation through nonlinear transformations. The projection process encompasses linear transformation, layer normalization, activation functions, and dropout operations, defined as follows:

$$\mathbf{Z}_0 = \text{Dropout}(\text{ReLU}(\text{LayerNorm}(\mathbf{W}_{\text{in}}\mathbf{X} + \mathbf{b}_{\text{in}}))) \quad (1)$$

where, $\mathbf{W}_{\text{in}} \in \mathrm{R}^{D \times d_{\text{model}}}$, $\mathbf{b}_{\text{in}} \in \mathrm{R}^{d_{\text{model}}}$ represent the learnable parameters of the linear transformation. Layer normalization (LayerNorm) is defined as follows:

$$\text{LayerNorm}(\mathbf{x}) = \gamma \cdot \frac{\mathbf{x} - \mu}{\sqrt{\sigma^2 + \grave{o}}} + \beta \quad (2)$$

where, μ and σ^2 denote the mean and variance, respectively, $\gamma, \beta \in \mathrm{R}^{d_{\text{model}}}$ are learnable parameters, and $\grave{o}$ is a small constant to prevent division by zero. The ReLU activation function introduces nonlinearity, enhancing the model's expressive capacity. Dropout randomly zeros elements with probability (p_{dropout}), mitigating overfitting. Layer normalization standardizes the feature distribution, alleviating internal covariate shift and improving training stability. Together, the ReLU activation function and dropout operations enhance the model's nonlinear modeling capability and generalization performance.

B. Position Coding Transformer

The Transformer architecture is inherently insensitive to the order of input tabular data, necessitating the explicit addition of positional information to preserve the sequential characteristics of the data. To retain the order information of tabular data features, this study employed sine and cosine functions to generate positional encodings, defined as:

$$\text{PE}(t, 2i) = \sin\left(\frac{t}{10000^{2i/d_{\text{model}}}}\right), \quad \text{PE}(t, 2i+1) = \cos\left(\frac{t}{10000^{2i/d_{\text{model}}}}\right) \quad (3)$$

where, $t \in \{0, 1, \ldots, \text{max_len} - 1\}$ denotes the feature step, and $i \in \{0, 1, \ldots, (d_{\text{model}}/2) - 1\}$ represents the dimension index. max_len is the maximum feature length. The positional encoding is added to the projected input, followed by a dropout operation, defined as follows:

$$\mathbf{Z}_1 = \text{Dropout}(\mathbf{Z}_0 + \mathbf{PE}_{[:T]}) \quad (4)$$

where, $\mathbf{PE}_{[:T]} \in \mathrm{R}^{T \times d_{\text{model}}}$ denotes the positional encoding truncated to the first T rows. The sine and cosine functions generate periodic signals, enabling the model to distinguish between different features while maintaining smooth frequency variations. Fixed positional encodings, as opposed to learnable encodings, reduce the number of parameters and offer better generalization across tabular data of varying lengths. The dropout operation further enhances model robustness, preventing over-reliance on positional encodings.

C. Transformer Encoder

The Transformer encoder serves as the core component of the model, tasked with capturing complex dependencies among features in tabular data. It consists of L stacked encoder layers, each comprising a multi-head self-attention mechanism and a feed-forward neural network (FFN), augmented by residual connections and layer normalization. For the l-th layer, given the input ($\mathbf{Z}_{l-1} \in \mathrm{R}^{B \times T \times d_{\text{model}}}$), the multi-head self-attention is computed as follows:

$$\mathbf{A}_h = \text{Softmax}\left(\frac{\mathbf{Q}_h \mathbf{K}_h^{\cdot}}{\sqrt{d_k}}\right)\mathbf{V}_h \quad (5)$$

Here, ($\mathbf{Q}_h = \mathbf{Z}_{l-1}\mathbf{W}_{Q,h}, \mathbf{K}_h = \mathbf{Z}_{l-1}\mathbf{W}_{K,h}, \mathbf{V}_h = \mathbf{Z}_{l-1}\mathbf{W}_{V,h}$) represent the query, key, and value matrices, respectively, $\mathbf{W}_{Q,h}, \mathbf{W}_{K,h}, \mathbf{W}_{V,h} \in \mathrm{R}^{d_{\text{model}} \times d_k}; d_k = d_{\text{model}}/n_{\text{head}}; n_{\text{head}}$ denotes the number of attention heads, $\sqrt{d_k}$ is the scaling factor used to stabilize gradient computations, and $h \in \{1, \ldots, n_{\text{head}}\}$ represents the multi-attention head. The multi-head self-attention mechanism enables the model to focus on different parts of the sequence in parallel, capturing diverse feature dependency patterns. Residual connections preserve the original input information, mitigating gradient vanishing issues in deep networks. The feed-forward neural network (FFN) applies a nonlinear transformation independently to each time step, enhancing feature representation, and is defined as follows:

$$\mathbf{F} = \text{ReLU}(\mathbf{Z}_l^{'}\mathbf{W}_1 + \mathbf{b}_1)\mathbf{W}_2 + \mathbf{b}_2 \quad (6)$$

Here, $\mathbf{W}_1 \in \mathrm{R}^{d_{\text{model}} \times d_{\text{ff}}}, \mathbf{W}_2 \in \mathrm{R}^{d_{\text{ff}} \times d_{\text{model}}}, \mathbf{b}_1 \in \mathrm{R}^{d_{\text{ff}}}, \mathbf{b}_2 \in \mathrm{R}^{d_{\text{model}}}, d_{\text{ff}}$ denotes the dimension of the feed-forward neural network. The FFN enhances the model's nonlinear modeling capability, capturing complex data patterns. Residual connections and layer normalization further improve training stability, enabling the stacking of additional layers to increase the model's capacity.

D. Classification Head

The classification head employs a MultiLayer Perceptron (MLP) to map the pooled feature vectors to a probability distribution over activity categories, generating the final predictions for physical activities. Specifically, the classification head first applies a global pooling operation, such as mean pooling or max pooling, to the sequence features output by the Transformer encoder, extracting representative global features. These features are then fed into a multilayer perceptron consisting of multiple fully connected layers, each incorporating nonlinear activation functions (e.g., ReLU) and dropout operations to enhance the model's expressive power and generalization performance. Finally, a softmax function is applied to map the output to a probability distribution over the activity categories, enabling precise classification of physical activities.

$$\mathbf{Y} = \text{Softmax}(\text{Dropout}(\text{ReLU}(\text{LayerNorm}(\mathbf{Z}_{\text{pool}}\mathbf{W}_3 + \mathbf{b}_3)))\mathbf{W}_4 + \mathbf{b}_4) \quad (7)$$

Here, $\mathbf{W}_3 \in \mathrm{R}^{d_{\text{model}} \times d_{\text{ff}}}, \mathbf{W}_4 \in \mathrm{R}^{d_{\text{ff}} \times C}, \mathbf{b}_3 \in \mathrm{R}^{d_{\text{ff}}}, \mathbf{b}_4 \in \mathrm{R}^{C}$. The deep classification head enhances the model's classification capability through additional linear layers, layer normalization, and nonlinear activation functions. Dropout operations effectively mitigate overfitting, improving the model's generalization performance on test data. The softmax function

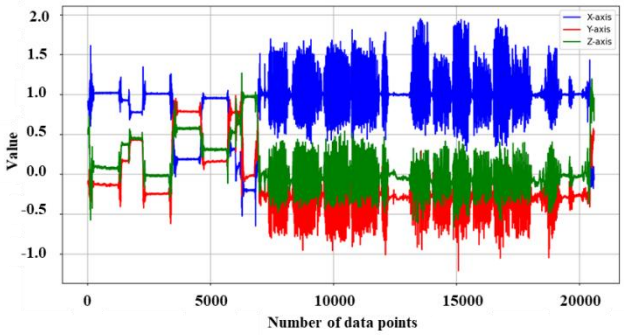
ensures that the output forms a valid probability distribution, making it suitable for multi-class classification tasks.

III. Analysis of Human Activity Perception Data

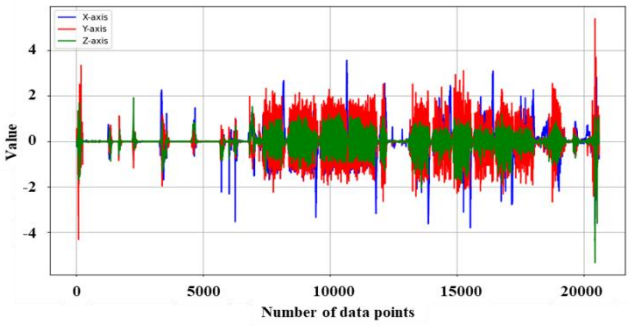
The proposed method facilitates efficient and accurate human activity recognition. The HAPT (Human Activity Recognition Using Smartphones) dataset [26], a widely adopted benchmark in HAR research, and available through the UCI Machine Learning Repository, captures diverse daily activities using smartphone-embedded inertial sensors. The dataset comprises time-series signals collected at 50 Hz from a waist-mounted smartphone, including: triaxial accelerometer data (X/Y/Z axes) quantifies linear acceleration patterns associated with body movements such as walking, sitting, or transitioning between postures. Triaxial gyroscope data (X/Y/Z axes) measures angular velocities to characterize rotational dynamics during activities like turning or limb motion. This multimodal sensor fusion provides complementary kinematic representations, enabling robust activity classification while maintaining temporal granularity critical for motion analysis. The standardized sampling protocol ensures consistent signal resolution across all recorded activities. The dataset for this study was collected from 30 volunteers aged between 19 and 48 years. During the experiments, a Samsung Galaxy S II smartphone was securely placed at the waist of each participant. This specific placement was chosen to optimize the capture of core trunk movements, thereby facilitating a more accurate recognition of whole-body activities. Participants were instructed to perform 12 predefined daily activities, each explicitly labeled. Fig. 2 illustrates the triaxial accelerometer and triaxial gyroscope signals recorded from one participant during an activity. The raw signals were preprocessed for noise reduction using a median filter and a third-order low-pass Butterworth filter with a cutoff frequency of 20 Hz. The clarity and distinctness of these signals demonstrate the smartphone's capability to accurately and effectively capture human motion data during activities. This provides compelling evidence for the feasibility of using smartphones as a robust tool for human activity recognition.

The complete dataset comprises a substantial volume of sensor readings. Following meticulous preprocessing and segmentation, these readings are transformed into a large collection of time-series samples. Data segmentation involves dividing continuous sensor data into fixed-length windows, typically 2.56 seconds (equivalent to 128 sampling points). Overlapping between windows is often employed to ensure each window comprehensively captures complete activity information. During the feature extraction phase, a diverse set of time-domain and frequency-domain features is extracted from the raw accelerometer and gyroscope signals for each 2.56-second window. These features are specifically designed to encapsulate the unique patterns associated with different human activities.

Following the crucial feature extraction step, each individual time window is successfully transformed into a high-dimensional feature vector. For instance, in the HAPT dataset, each such sample is represented by 561 carefully selected features, and these high-dimensional vectors subsequently serve as input data for training and inference with various machine learning models. To offer a more intuitive perspective, allowing for a clear understanding of the unique distribution patterns of these feature values across different activity types. Fig. 3 illustrates the 561 feature values for a single sample within a specific activity class. From the visual representation in this figure, it is evident that all 12 distinct human activity types, including walking, ascending stairs, descending stairs, sitting, standing, lying, and a range of complex transitional movements (such as standing-to-sitting, sitting-to-standing, sitting-to-lying, lying-to-sitting, standing-to-lying, and lying-to-standing), exhibit notably distinct feature curves. Specifically, when the human body is engaged in dynamic physical activities, the extracted feature values tend to be quantitatively higher. Conversely, when the body is in a static or non-moving state, such as sitting or lying, the corresponding feature values are generally observed to be at lower levels. A particularly interesting phenomenon is observed during standing posture, where the distribution of feature values demonstrates an intriguing duality: approximately half of the feature values display relatively high magnitudes, while the other half are comparatively smaller. Although these differences in feature values indeed provide a basis for distinguishing between various activity types and demonstrate a certain discriminatory capability, recognizing the inherent and significant inter-individual variability in behavioral characteristics, we acknowledge the limitations of existing methods. Therefore, future research urgently needs to develop more advanced and intelligent methods and algorithms to achieve more accurate and faster recognition and classification of human physical activity types, thereby better adapting to the complex and diverse patterns of human movement in real-world scenarios.

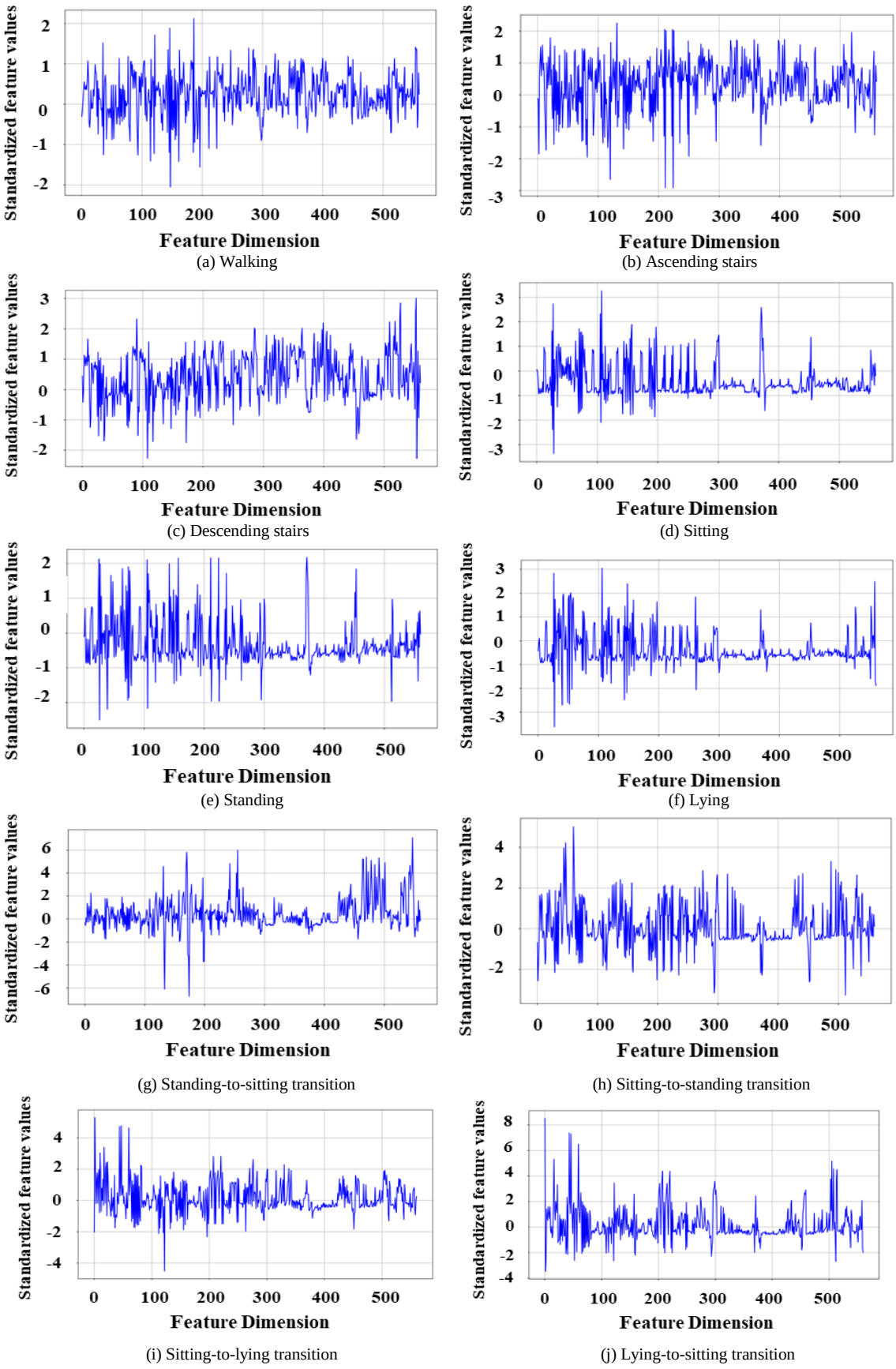


(a) Variation curves of the three-axis acceleration (m/s2).



(b) Variation curves of the three-axis gyroscope (rad/s).

Fig. 2. Two types of perception signal curves of a certain tester.



(a) Walking

(b) Ascending stairs

(c) Descending stairs

(d) Sitting

(e) Standing

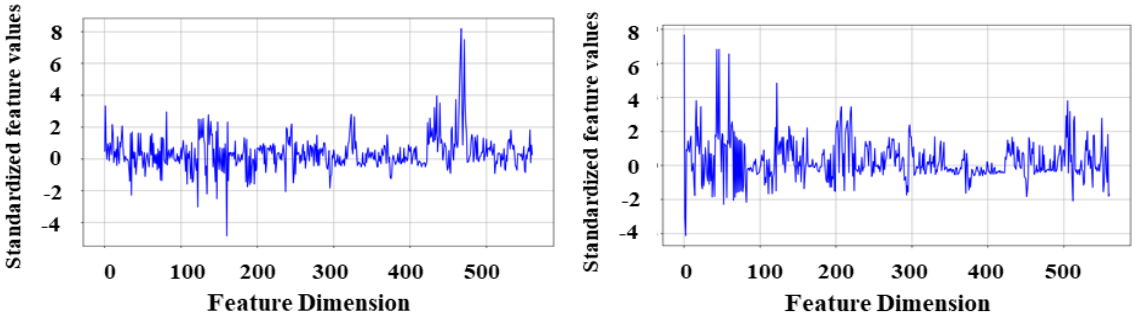
(f) Lying

(g) Standing-to-sitting transition

(h) Sitting-to-standing transition

(i) Sitting-to-lying transition

(j) Lying-to-sitting transition



(k) Standing-to-lying transition (l) Lying-to-standing transition

Fig. 3. Display of characteristic curves of different types of activities.

The class distribution of the dataset utilized in this study is presented in Table I. The training set comprises 7,767 samples, while the test set contains 3,153 samples, yielding a split ratio of approximately 2.46:1. This partitioning is considered reasonable and aligns with conventional machine learning data splitting standards, such as 70%-30% or 80%-20% distributions. Within the dataset, Classes 1 to 6 are identified as major categories, exhibiting substantial sample sizes. Specifically, each of these classes contains over 1,000 samples in the training set and more than 400 samples in the test set, collectively constituting the vast majority of the data. Conversely, Classes 7 to 12 are designated as minority categories, characterized by extremely sparse sample representation. For instance, the training set for minority classes contains no more than 90 samples each, and the test set no more than 50 samples each. Notably, Class 8 is the most underrepresented, with only 23 samples in the training set and 10 samples in the test set, indicating a severe class imbalance issue within the dataset.

To elaborate, the largest class, Class 5, contains 1,423 samples in the training set, which is 62 times larger than the smallest class, Class 8, with only 23 samples; a similar disparity is observed in the test set. Furthermore, the combined total of minority classes (Classes 7 to 12) accounts for only 4.5% of the total samples in the training set and 5.3% in the test set. This extreme imbalance implies that conventional evaluation metrics, such as overall accuracy, may be predominantly influenced by the majority classes, thereby potentially masking performance deficiencies on the minority classes. It is worth noting that the proportion of samples for each class is largely consistent between the training and test sets. For instance, Class 1 accounts for 15.8% of the training set and 15.7% of the test set. This consistency suggests that the data partitioning maintained distributional uniformity, effectively mitigating potential biases introduced by the split. Nevertheless, the dataset presents a small-class risk: certain minority classes, such as Classes 8, 10, and 12, possess extremely low sample counts in the test set (no more than 27 samples each), which could lead to significant volatility in evaluation results. Overall, the dataset exhibits a pronounced long-tail distribution, where the insufficient number of samples in minority classes may constrain the model's generalization capabilities.

TABLE I. ANALYSIS OF THE CLASS DISTRIBUTION OF THE DATASET

Class	Train dataset	Test dataset
1	1226	496
2	1073	462
3	987	420
4	1293	508
5	1423	556
6	1413	545
7	47	23
8	23	10
9	75	32
10	60	25
11	90	49
12	57	27
Total	7767	3153

IV. COMPARATIVE TESTING AND ANALYSIS

A. Model Training and Evaluation

In order to fully verify the accuracy and feasibility of the proposed method, human perception data is used to update the constructed recognition network. In addition, three other classic machine learning algorithms were used for comparison. The assessment results are shown in Table II. All evaluated models demonstrated excellent performance on the training set, clearly indicating their strong ability to fit the training data. Specifically, Random Forest achieved 100% accuracy, precision, recall, and F1-Score on the training set. While its training performance was perfect, a significant drop in performance on the validation set strongly suggests severe overfitting. In contrast, Long Short-Term Memory (LSTM) networks, and vanilla Recurrent Neural Networks (RNNs) maintained metrics between 97% and 98% on the training set, performing well but slightly below Random Forest. Of particular note, Transformer2 (employing a class-balancing strategy) exhibited near-perfect performance on the training set, with an accuracy of 99.95%. This is likely attributable to the adopted class-balancing strategies, such as oversampling or weighted loss, which enabled the model to better learn and recognize minority classes, thereby improving overall training effectiveness.

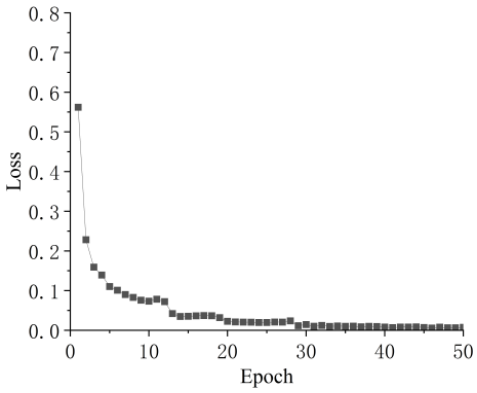
TABLE II. COMPARISON OF THE EVALUATION EFFECTS OF DIFFERENT RECOGNITION ALGORITHMS

Methods	Training process				Validation process			
	Accuracy	*Precision*	*Recall*	*F1-Score*	*Accuracy*	*Precision*	*Recall*	*F1-Score*
Random forest	**1**	**1**	**1**	**1**	**0.9083**	**0.9095**	**0.9083**	**0.9074**
LSTM	**0.9799**	**0.9810**	**0.9799**	**0.9799**	**0.9230**	**0.9247**	**0.9230**	**0.9226**
Vanilla RNN	**0.9813**	**0.9819**	**0.9813**	**0.9813**	**0.9241**	**0.9273**	**0.9241**	**0.9232**
Transformer	**0.9776**	**0.9784**	**0.9776**	**0.9766**	**0.9247**	**0.9280**	**0.9247**	**0.9238**
Transformer2 (class-balanced)	**0.9995**	**0.9995**	**0.9995**	**0.9995**	**0.9397**	**0.9404**	**0.9397**	**0.9396**

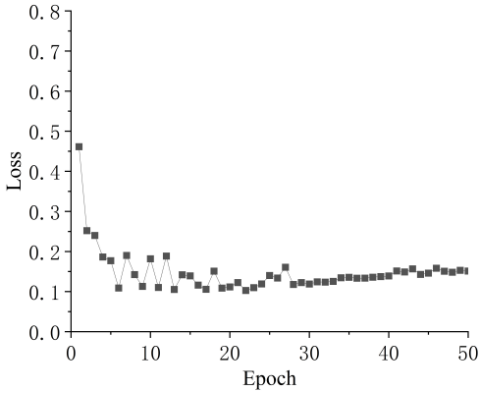
Performance on the validation set more accurately reflects a model's generalization capability—its ability to handle unseen data. Random Forest's performance significantly declined on the validation set, with an accuracy of only 90.83%, a stark contrast to its 100% training performance. This further confirms its severe overfitting, rendering the model unsuitable for direct real-world application. The LSTM model performed relatively well on the validation set, achieving an accuracy of 93.30%, outperforming vanilla RNN. This indicates that LSTMs have an advantage in processing sequential data, particularly in capturing long-term dependencies. Vanilla RNN model showed comparable performance on the validation set, with accuracies around 92.4% to 92.5%, but slightly trailed LSTM. Vanilla RNNs' performance might be limited by long-range dependency issues, while the standard Transformer's optimization might be insufficient with limited data, preventing it from fully realizing its potential in sequential data processing. Most notably, Transformer2 (class-balanced) achieved the best performance on the validation set, with an accuracy of 93.97%. Furthermore, this model also showed the smallest performance gap between the training and validation sets, dropping from 99.95% to 93.97%. This result strongly demonstrates the effectiveness of the class-balancing strategy, as it not only significantly mitigated overfitting but also substantially enhanced the model's generalization capability, allowing it to handle imbalanced datasets more effectively.

In summary, while Random Forest achieved 100% accuracy on the training set, it only managed 90.83% on the validation set, indicating severe overfitting. This might be due to the tree-based model's oversensitivity to noise in the training data. The Transformer2 (class-balanced) model demonstrated the best generalization capability, exhibiting the smallest performance gap between its training and validation sets. This highlights the crucial role of class-balancing strategies in improving a model's generalization performance on imbalanced datasets. Among other sequential models, LSTM outperformed both vanilla RNN and the standard Transformer, suggesting that on moderately sized datasets, LSTM's sequential modeling capabilities still hold a significant advantage. Furthermore, this study clearly indicates that class-balancing strategies can significantly enhance the Transformer model's generalization performance, especially in handling minority classes more favorably and robustly.

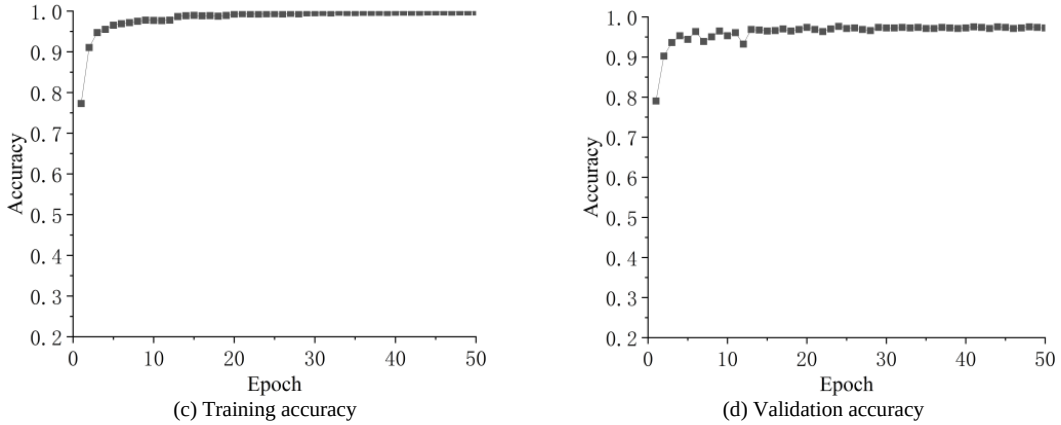
To provide a more intuitive understanding of the training and validation processes of the proposed method, relevant evaluation curves are presented in Fig. 4. As depicted in the loss curve shown in Fig. 4(a) and Fig. 4(b), the training loss value decreases rapidly with an increasing number of training epochs. By approximately 30 training epochs, the training loss value has essentially approached zero and remains highly stable thereafter. The validation loss also gradually stabilizes around 0.1 at approximately 30 training epochs. Although some fluctuations are observed subsequently, the validation loss consistently remains within the vicinity of 0.15. This indicates that the model's fit on the validation set has reached a relatively stable state.



(a) Training loss



(b) Validation loss



(c) Training accuracy (d) Validation accuracy

Fig. 4. Training and validation process of the proposed method.

Further analysis of the training accuracy curve in Fig. 4(c) reveals that the model's accuracy on the training set exceeded 0.95 when the training epochs were merely 10. It then continued to improve, gradually stabilizing and ultimately approaching 1. Similarly, as shown in the validation accuracy curve in Fig. 4(d), the model's accuracy on the validation set reached approximately 0.95 at just 10 training epochs. Despite slight subsequent fluctuations, the validation accuracy consistently remained stable around 0.95. Overall, these curves collectively demonstrate that the method proposed in this study exhibits rapid convergence in the early stages of training and achieves high performance levels on both the training and validation sets. The swift convergence of training loss to near zero, coupled with the stable fluctuations of validation loss, indicates that while the model effectively learns features from the training data, it also possesses a commendable degree of generalization capability. The consistent stability of the validation accuracy around 0.95 further corroborates the model's robustness and effectiveness.

V. Discussion and Analysis

To provide a more comprehensive analysis of the strengths and weaknesses of the proposed method, we conducted a detailed calculation of evaluation metrics for each activity class within the test set. The assessment results are shown in Table III. This approach effectively illustrates where the proposed method excels in recognition and where its performance is suboptimal. Overall, the classification model demonstrates excellent performance on the majority of classes, particularly those with higher support, but exhibits instability on categories with lower support, revealing a clear dependency on data availability.

The proposed method performs exceptionally well on high-support categories, namely Classes 1 to 6. These classes possess a relatively large number of samples, with each class having a support of over 400 samples. Across these activities, the model's precision, recall, and F1-Score are generally high. Notably, Classes 1, 2, 5, and 6 all achieved F1-Scores exceeding 90%, with Class 6 performing best at an F1-Score of 98.32%, indicating near-perfect classification. However, for Class 3, despite an exceptionally high precision of 98.88%, its recall was relatively low at 84.05%. This suggests that the model is highly strict in its predictions for this class, potentially leading to a certain degree of missed detections. To balance precision and recall, future adjustments to the classification threshold could be considered to improve recall for this category.

TABLE III. Evaluation Effect of Each Type of Human Activity

Class	Precision	Recall	F1-Score	Support degree
1	0.9215	0.9940	0.9564	496
2	0.8920	0.9654	0.9272	462
3	0.9888	0.8405	0.9086	420
4	0.9287	0.8720	0.8995	508
5	0.8798	0.9478	0.9126	556
6	1	0.9670	0.9832	545
7	0.8421	0.6957	0.7619	23
8	0.7692	1	0.8696	10
9	0.6786	0.5938	0.6333	32
10	0.7619	0.6400	0.6957	25
11	0.6875	0.6735	0.6804	49
12	0.6552	0.7037	0.6786	27

In contrast to the high-support categories, the proposed method's performance on low-support categories, namely Classes 7 to 12, was relatively poor and unstable. These classes have fewer samples, with each class having a support of less than 50 samples, leading to significant fluctuations in the model's evaluation metrics. For instance, in Class 8, although the model's recall was remarkably high at 100%, indicating its ability to capture all true positive samples, its precision was only 76.92%. This suggests a high false positive rate, where the model incorrectly identifies some samples as belonging to Class 8 when they do not. Furthermore, Class 9 exhibited low precision and recall, at 67.86% and 59.38% respectively, resulting in an F1-Score of 63.33%, the lowest among all categories. This clearly indicates that the model's discriminative ability for Class 9 is relatively weak. This instability primarily stems from insufficient data, as the model was unable to adequately learn the crucial features of these minority classes, leading to reduced generalization capabilities.

It is important to note that some categories, such as Classes 3, 8, and 12, exhibit an imbalance between precision and recall. This suggests that the model's classification thresholds may require optimization. Specifically, for categories with high precision but low recall (e.g., Class 3), the classification threshold could be appropriately lowered to improve recall while maintaining reasonable precision. Conversely, for categories with high recall but low precision (e.g., Class 8), the classification threshold could be raised to reduce false positives and enhance precision. While the model demonstrates excellent performance on data-rich categories, there remains significant room for improvement in its recognition effectiveness for small-sample categories. Future optimization efforts should focus on increasing the data volume for minority classes and further refining the model's classification strategies, for example, by adjusting classification thresholds or employing cost-sensitive learning. These measures aim to comprehensively enhance the model's overall performance, particularly its robustness in handling imbalanced datasets.

From the above analysis, it can be seen that the method proposed in this study combines the data balance strategy with the Transformer model. Compared with traditional machine learning methods, it shows significant advantages in recognition accuracy. Although Reshmi et al. [27] proposed a recognition method based on feature selection, their work did not fully consider the impact of data imbalance on model performance. Pavliuk et al. [28] focused on exploring the application of transfer learning in the task of human activity recognition and did not delve deeply into the issue of unbalanced category distribution either. In fact, the problem of data imbalance is widely present in sensor-based activity recognition tasks, which can easily lead to the model being biased towards the majority of classes and a decline in generalization ability. Therefore, it must be fully considered in the method design stage. This study effectively alleviates this problem by introducing a balance strategy, thereby enhancing the robustness and recognition accuracy of the model in real scenarios.

VI. Conclusion

This study proposes a deep learning-based smartphone physical activity recognition framework, designed to address the prevalent decline in physical activity among college students. The framework's innovation lies in its utilization of high-precision Inertial Measurement Units (IMUs) embedded in smartphones to capture multi-dimensional real-time body postures during daily activities. It then employs an advanced Transformer architecture as the core classifier. The Transformer's unique Self-Attention Mechanism effectively extracts complex spatiotemporal features from sensor data, enabling precise recognition of various physical activities (such as walking, running, and climbing stairs). This integrated approach of intelligent sensing and advanced algorithms provides a reliable and efficient technological solution for monitoring college students' physical activity.

1) After achieving 99.95% on the training set, the Transformer2 (class-balanced) model maintained an accuracy rate of 93.97% on the validation set. The minimum performance gap between the training set and the validation set strongly demonstrates its excellent generalization performance and ability to handle imbalanced datasets.

2) The recognition accuracy rate of the Transformer2 (Class Balance) model is 93.97%, which is superior to 93.30% of LSTM, approximately 92.4%-92.5% of vanilla RNN and standard Transformer, and 90.83% of Random Forest.

3) The recognition accuracy of high-support categories is very high, but low-support categories remain challenging. Our model achieves extremely high recognition accuracy for highly supported categories (categories 1-6, with over 400 samples in each category). However, for the low-support categories (7-12 categories, with less than 50 samples for each category), the performance of the model fluctuates greatly.

4) The class balance strategy effectively enhances the robustness of the Transformer model. Although this numerical improvement seems small, it indicates that the model is more robust when dealing with minority classes, significantly reduces overfitting caused by data imbalance, and improves the overall generalization performance.

5) This model has a fast convergence speed and stable verification accuracy in the early stage of training. This indicates that the model has efficient learning ability and good stability, and can achieve a high-performance level within a relatively short training time.

The proposed method remains dependent on data augmentation or supplementary samples to improve recognition performance for minority classes, without fundamentally resolving the underlying data scarcity challenge. Although the class-balancing strategy enhances model robustness when handling minority categories, the limited magnitude of overall performance improvement indicates that the current approach still exhibits insufficient efficacy in addressing severe class imbalance scenarios, suggesting significant potential for further refinement. Future research will focus on further improving the model's recognition performance for low-support activity categories. Explore more advanced data augmentation technologies to expand the sample of ethnic minorities. The meta-learning or Few-Shot learning methods are studied to enable effective learning even under sparse data. Additionally, we will consider extending this framework to more complex body activity pattern recognition and exploring its potential applications in long-term health monitoring and personalized health intervention.

Acknowledgment

The authors gratefully acknowledge the support from the 2022 Liaoning Provincial Natural Science Foundation Program Key Science and Technology Innovation Base Joint Open Fund(2022-KF-18-04), Shanghai Municipal Commission of Education (C2024090) and Scientific research project of Shanghai Vocational College of Agriculture and Forestry (KY(6)2-0000-23-13).

References

[1] S. Majumder, and M. Deen, “Smartphone sensors for health monitoring and diagnosis”, Sensors, vol. 19, pp. 216, 2019. DOI: 10.3390/s19092164

[2] H. Sarmadi, A. Entezami, K. V. Yuen, and B. Behkamal, "Review on smartphone sensing technology for structural health monitoring". Measurement, vol. 223, pp. 113716, 2023. DOI: 10.1016/j.measurement.2023.113716

[3] Y. G. Lee, W. S. Jeong, and G. Yoon, "Smartphone-based mobile health monitoring", Telemedicine and e-Health, vol. 18, pp. 585-590, 2012. DOI: 10.1089/tmj.2011.0245

[4] H. F. Nweke, Y. W. Teh, G. Mujtaba, and M. A. Al-Garadi, "Data fusion and multiple classifier systems for human activity detection and health monitoring: Review and open research directions", Information Fusion, vol. 46, pp. 147-170, 2019. DOI: 10.1016/j.inffus.2018.06.002

[5] J. Yin, Q. Yang, and J. J. Pan, "Sensor-based abnormal human-activity detection", IEEE Transactions on Kowledge and Data Engineering, vol. 20, pp. 1082-1090, 2018. DOI: 10.1109/TKDE.2007.190662

[6] A. Saeed, T. Ozcelebi, and J. Lukkien, "Multi-task self-supervised learning for human activity detection". ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, vol. 61, p. 1-30, 2019. DOI: 10.1145/3328932

[7] M. Janidarmian, F. A. Roshan, K. Radecka, and Z. Zilic, "A comprehensive analysis on wearable acceleration sensors in human activity recognition", Sensors, vol. 17, pp. 529, 2017. DOI: 10.3390/s17030529

[8] S. Iloga, A. Bordat, J. Kernec, and O. Romain, "Human activity recognition based on acceleration data from smartphones using HMMs", IEEE Access, vol. 9, pp. 139336-139351, 2021. DOI: 10.1109/ACCESS.2021.3118472

[9] M. B. Del Rosario, S. J. Redmond, and N. H. Lovell, "Tracking the evolution of smartphone sensing for monitoring human movement", Sensors, vol. 15, pp. 18901-18933, 2015. DOI: 10.3390/s150818901

[10] I. A. Faisal, T. W. Purboyo, and A. S. R. Ansori, "A review of accelerometer sensor and gyroscope sensor in IMU sensors on motion capture". ARPN Journal of Engineering and Applied Science, vol. 15, pp. 826-829, 2019. DOI: 10.36478/jeasci.2020.826.829

[11] I. M. Pires, N. M. Garcia, E. Zdravevski, and P. Lameski, "Daily motionless activities: a dataset with accelerometer, magnetometer, gyroscope, environment, and GPS data", Scientific Data, vol. 9, pp. 105, 2022. DOI: 10.1038/s41597-022-01213-9

[12] S. Hernandez Sanchez, R. Fernandez Pozo, and L. A. Hernandez Gomez, "Estimating vehicle movement direction from smartphone accelerometers using deep neural networks", Sensors, vol. 18, pp. 2624, 2018. DOI: 10.3390/s18082624

[13] X. Huang, Y. Xue, S. Ren, and F. Wang, "Sensor-based wearable systems for monitoring human motion and posture: A review", Sensors, vol. 23, pp. 9047, 2023. DOI: 10.3390/s23229047

[14] O. D. Lara, A. J. Pérez, M. A. Labrador, and J. D. Posada, "Centinela: A human activity recognition system based on acceleration and vital sign data. Pervasive and Mobile Computing", vol. 8, pp. 717-729, 2012. DOI: 10.1016/j.pmcj.2012.06.004

[15] D. Micucci, M. Mobilio, and P. Napoletano, "Unimib shar: A dataset for human activity recognition using acceleration data from smartphones", Applied Sciences, vol. 7, pp. 1101, 2017. DOI: 10.3390/app7101101

[16] M. O. Mario, "Human activity recognition based on single sensor square HV acceleration images and convolutional neural networks". IEEE Sensors Journal, vol. 19, pp. 1487-1498, 2018. DOI: 10.1109/JSEN.2018.2882943

[17] W. Niu, J. Long, D. Han, and Y. Wang, "Human activity detection and recognition for video surveillance", In 2004 IEEE International Conference on Multimedia and Expo (ICME)(IEEE Cat. No. 04TH8763), Taipei, pp. 719-722, 2014. DOI: 10.1109/ICME.2004.1394293

[18] J. Sung, C, Ponce, B. Selman, and A. Saxena, "Unstructured human activity detection from rgbd images", In 2012 IEEE International Conference on Robotics and Automation, Saint Paul, p. 842-849, 2012. DOI: 10.1109/ICRA.2012.6224591

[19] B. Ni, Y. Pei, P. Moulin, and S. Yan, "Multilevel depth and image fusion for human activity detection", IEEE Transactions on Cybernetics, vol. 43, pp. 1383-1394, 2013. DOI: 10.1109/TCYB.2013.2276433

[20] H. Koppula, and A. Saxena, "Learning spatio-temporal structure from rgb-d videos for human activity detection and anticipation", In Proceedings of the 30th International Conference on Machine Learning, Atlanta, pp. 792-800, 2013. https://proceedings.mlr.press/v28/koppula13.html

[21] M. Albanese, R. Chellappa, V. Moscato, A. Picariello, V. S. Subrahmanian, P. Turaga, and O. Udrea, "A constrained probabilistic petri net framework for human activity detection in video", IEEE Transactions on Multimedia, vol. 10, pp. 1429-1443, 2008. DOI: 10.1109/TMM.2008.2010417

[22] R. Rodrigues, N. Bhargava, R. Velmurugan, and S. Chaudhuri, "Multi-timescale trajectory prediction for abnormal human activity detection", In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Snowmass, pp. 2626-2634, 2022. DOI: 10.1109/WACV45572.2020.9093633

[23] F. M. Shiri, T. Perumal, N. Mustapha, and R. Mohamed, "A comprehensive overview and comparative analysis on deep learning models", Journal on Artificial Intelligence, vol. 6, pp. 301-360, 2024. DOI: 10.32604/jai.2024.054314

[24] Z. Niu, G. Zhong, and H. Yu, "A review on the attention mechanism of deep learning", Neurocomputing, vol. 452, pp. 48-62, 2021. DOI: 10.1016/j.neucom.2021.03.091

[25] K. Han, A. Xiao, E. Wu, J. Guo, C. Xu, and Y. Wang, "Transformer in transformer", Advances in Neural Information Processing Systems, vol. 34, pp. 15908-15919, 2021. https://arxiv.org/abs/2103.00112.

[26] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones", In ESANN 2013 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, Belgium, pp. 437-442, 2013.

[27] S. Reshmi, & E. Ramanujam, "An ensemble maximal feature subset selection for smartphone based human activity recognition", Journal of Network and Computer Applications, vol. 226, pp. 103875, 2024. DOI: 10.1016/j.jnca.2024.103875

[28] O. Pavliuk, M. Mishchuk, and C. Strauss, "Transfer learning approach for human activity recognition based on continuous wavelet transform", Algorithms, vol. 16, no.2, pp. 77, 2023. DOI: 10.3390/a16020077

Game Theory-Optimized Attention-Based Temporal Graph Convolutional Network for Spatiotemporal Forecasting of Sea Level Rise

T M Swathy[1], K.Ruth Isabels[2], Dr. A. Sindhiya Rebecca[3], Venubabu Rachapudi[4],
Prof. Ts. Dr. Yousef A.Baker El-Ebiary[5], Dr. Shobana Gorintla[6], Elangovan Muniyandy[7]
Assistant Professor, Dept. of Electrical Engineering, Rungta College of Engineering and Technology, Bhilai, India[1]
Associate Professor, Dept. of Mathematics, Saveetha Engineering College (Autonomous), Thandalam, Chennai, Tamilnadu, India[2]
Assistant Professor, Department of Master of Business Administration, Panimalar Engineering College, Chennai, India[3]
Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India -522302[4]
Faculty of Informatics and Computing, UniSZA University, Malaysia[5]
Associate Professor, Department of CSE, NRI Institute of Technology, Pothavarapadu, Agiripalli mandal Pincode-52120, Krishna District, Andhra Pradesh, India[6]
Department of Biosciences, Saveetha School of Engineering. Saveetha Institute of Medical and Technical Sciences, Chennai - 602 105, India[7]
Applied Science Research Center. Applied Science Private University, Amman, Jordan[7]

***Abstract*—Predicting Sea level rise accurately is crucial in the formulation of effective adaptation plans to counteract the effects of climate change in vulnerable coastal areas, infrastructure, and people. The conventional forecasting models tend to fail in capturing the intricate spatiotemporal relationships affecting sea level variations. In order to overcome the above-mentioned challenges, this research introduces a hybrid predictive model combining a Temporal Graph Convolutional Network (T-GCN) with attention and game theory-based optimization strategy. T-GCN structure is specially tailored to capture spatial dependencies as well as temporal dynamics in sea level change, providing even deeper understanding of the changing dynamics of sea levels. The attention mechanism strengthens the model by dynamically weighing important variables, whereas the game-theoretic optimization efficiently optimizes multiple objectives, e.g., prediction accuracy and robustness. Experimental results, measured in terms of common performance indicators, show the better effectiveness of the proposed model with a correlation coefficient of 0.996512 and an overall error of 0.032154. Through the inclusion of both climatic and socio-economic variables, this methodology provides accurate, data-based insights to inform climate policy and adaptive planning. The results highlight the capabilities of state-of-the-art machine learning methods for solving actual sea level rise challenges.**

Keywords*—*Temporal graph convolutional networks; attention mechanisms; game theory optimization; sea level rise prediction; climate change adaptation

I. Introduction

Melting of ice possesses a creeping threat of global significance with potential impacts and effects on any coastal area, population, structures, and ecosystems [1], [2]. Since the late 80s, the climate of the earth has been changing because of various anthropogenic factors for instance emission of greenhouse gases and deforestation and as expected the polar ice caps and the glaciers are melting [3], [4]. This phenomenon is now becoming a severe danger to millions of the lowland people, especially those in developing and overpopulated coastal zones. These several impacts of sea level rise are diverse: there are storms and their consequences such as frequent and severe floods, more powerful storm surges and also rapidly progressing coastal erosion [5]. While such changes in physical environment can result in loss of people's homes, loss of animal and plant habitats, destruction of valuable ecosystems like mangrove forests and corals that offer protection against storms [6]. In addition, sectors that depend on the coasts and sea, for instance, fishery, tourism, and shipping business experiences difficulties and higher expense charges. Most of the designed infrastructure of roads and bridges, sewage systems without which no urban development is complete, require heavier investment and modifications to withstand the prevailing and projected climate change [7].

However, when it comes to predicting sea level increase something that is so important and mandatory in the contemporary world, the traditional models fail to address the issue in a proper way. Sea level fluctuation depends on the number of factors which are interrelated and interconnected: pressure at sea level, currents at sea level, and tectonic activity [8]. These interactions produce a very complex environment that appears as being programmed and organized in which shifting one factor can influence other related factors in a rather unpredictable manner [9]. For instance, changes in pressure may affect currents and the circulation of the ocean, and movements in the earth's crust may change sea level through vertical movements of the land mass [10]. Issues concerning flexibility, capacity utilization, uncertainties and seasonality are some of the complexities that the conventional methods cannot capture

easily since they depend mostly on the past data trends, which have a linear relationship. Thus, they give predictions that are rather too simplistic and can be quite off base, underestimating or even overestimating dangers that might come with the rise in sea level. Mainstream statistical and physical approaches fail to adequately capture a relationship between these factors because the typical applied parameters are not specific enough to capture the differences of coastal areas [11]. Coastal sea level management entails the physical alterations, proposed structures, and noticed events to reflect the response of the interested groups, such as the residents, business holders, ecological nature lovers, and policy makers. Both of these are crucial in comprehending the socioeconomic and cultural effects of sea level changes as well as in creating effective and fair response measures towards these changes. However, conventional models are mostly often distributed in terms of physical and environmentally related information devoid of more human factor aspects of the sea level [12].

1) Research motivation: CNN are specifically preferred for spatial pattern recognition, but in the event of a temporal sequence, their efficiency is greatly diminished with a result of creating likely errors in the forecasted values [13]. Originally, the T-GCN to analyse sea level rise are very efficient, where the data are described based on the graphs that nodes are used for spatial features while the edges focus on time features [14]. T-GCNs are less inclined towards computational efficiency, and sometimes it can cause scalability issues whenever dealing with big data sets. They also need to be fine-tuned due to the trade-off they create between the spatial and temporal dimensions, which can be difficult in some cases. In the other hand, it is useful to learn which features should be paid more attention, improve the model's temporal and spatial consideration. However, when the attention mechanism is applied separately, the overfitting occurs as it pays much more attention to the certain features while there might be other relevant features in some other context also equally important [15]. Moreover, attention mechanisms might not have the considerations of the spatial relation of the data which is important for sea level analysis. Such limitations call for the blended approach that use the strong characteristics of these methods while avoiding their related drawbacks. By connecting T-GCNs, attention mechanisms, the model developed in this article assumes to offer a more precise and complete solution for forecasting sea level rises.

2) Research significance: With the deficiency of conventional sea level rise prediction models, this paper suggests a new scheme that adopts game theory-based optimization, a self-attention mechanism, and T-GCN. T-GCNs are powerful in extracting intricate spatiotemporal information, and the self-attention mechanism allows the model to dynamically focus on the most relevant features. By adopting elements of game theory—mimicking strategic choice making and reconciling conflicting factors—the forecasting ability of the model is greatly improved. The combination of these methods leads to more robust, accurate, and actionable forecasts, providing valuable inputs to enable effective coastal adaptation planning.

3) Research contribution: The following are the contributions of this research;

a) This paper introduces a new hybrid model that integrates T-GCN, attention mechanisms, and game theory-based optimization to enhance the precision of sea level rise predictions.

b) The suggested framework efficiently describes spatial dependencies and temporal patterns from multivariate sea level datasets in a better manner than the traditional isolated deep learning techniques.

c) Attention mechanisms are used to dynamically emphasize and prioritize important spatial and temporal features, increasing interpretability of the model and concentrating attention on the most influential variables.

d) Game-theoretic approaches are employed to maximize trade-offs between two or more conflicting objectives—e.g., prediction performance and model resilience—and thus reduce overfitting and enhance generalizability overall.

The rest of the paper is organized as follows: Section II overviews related work. Section III presents the background to the study and formulation of the problem statement. Section IV outlines the method that has been proposed. In the Section V of the paper we present the experimental results and then compare them. Section VI provides the last and final conclusion of the paper and points out for the future work.

II. Related Work

According to Memarian Sorkhabi, Shadmanfar, and Al-Amidi [16] climate change was a global phenomenon and particularly the increase in sea level and frequency of floods had been recognized as one of the critical challenges for coastal cities. This study examined the temporal fluctuations of the sea-level which involves data like the SST originated from MODIS, wind speed and precipitation rates, and the change in the sea-level acquired through satellite altimetry. A weighted combination of values created a context vector that impacts the end output. T-GCNs model intricate spatial-temporal patterns, and attention boosted significance in features for accurate, area-based predictions in climate adaptation.lts and various forms of forecasting, which might not have taken into perspective all potential factors influencing the future environment. They could have been affected by the nature and resolution of the inputs used in making the prediction and there could be unknown climatic and socio-economic factors that could affected the sea level rise and floods on the resilience of cities on the coastline.

Accarino et al. [17] analyzed the effectiveness of LSTM neural network for short-term sea-level forecasting for the SANI region of the Mediterranean Sea. The work was described by multiple of models that used LSTM networks aimed at the three-day mean sea level forecast for a set of coastal regions. Concerning the evaluation of the projections, the latter were compared to observation data stemming from tide-gauge instruments and to the outputs from another model called SANIFS that had been designed at the Euro-Mediterranean Center on Climate Change. The incorporation of the LSTM-

based forecasting models into the Ophidia HPDA system, would open up a vast potential of adding capabilities of carrying large scale Time Series data and functioned in realization through application of HPC and Data science capabilities. The enhancement of these reasons could have resulted in greatly improved sea level prediction models in the future.

Prediction performance analysis carried out by Altunkaynak and Kartal [18] involved applying and comparing the results of various ML models based on sea level time series data obtained under different conditions. This study used DWT to analyze three stations in the Bosphorus Strait, and using SVM, k-NN, and DT to transfer sea level signal among the stations for their regions. The models applied were compared according to their performance in as far as predicting sea levels up to 7day in advance by subjecting them to RMSE and NSE tests. The findings showed that the developed models had higher competence in transferring information from one station to the next when the station is in the proximity as opposed to when stations were periodically close. One of the studies' drawbacks partly stemmed from its limited focus, which made it difficult to apply the findings to other areas or settings.

Raj [19] employed the first China's global ocean CDRs to evaluate and forecast changes in the Yellow Sea levels that significantly varied by season. With the help of the SSA method, it was suggested to analyze the characteristic and de-noise spatiotemporal and SLAs time series data in the Yellow Sea. The research then used SSA to build an SSA based model called SSA-LSTM for forecasting trends in sea level. 04 mm for the SLA time series prediction while the RMSE was found to be 19. 68 mm for the one-year spatiotemporal SLA forecast. There were also certain limitations in this research: The data set used in this research is limited to China's first global ocean CDRs and it only examined the Yellow Sea so the application of SSA-LSTM combined model would have been slightly different when studied on other areas with different ocean and climatology processes. Furthermore, the study does not explore any interference from other external factors, including the climatic fluctuations that always occur in the world, which may in one way or the other influence the sea levels, and this therefore undermined the strength and applicability of the forecast made in other or different setting.

Zhao, Cai, and Sun [20] discussed the contemporary issue of SLR that still posed a threat to small island states such as the Kiribati and Tuvalu among others; this attested to the need to enhance the reliability of data provided in advance of policy measure. The model was benchmarked against three other AI models: Some of the boosting technique out of which are adaboost, and multilinear regression. It was mentioned at the two places measures that it presents the least error rates. Applying, for instance, the trend analysis applied to gradual change, based on the linear regression, the GNSS-VLM-corrected long-term mean sea level increase is at 2. The rate of absolute sea level rises according to the estimated for Kiribati was 1mm/year while that of Tuvalu is 3mm/year.

Raj et al. [21] explained a novel technique for estimating sea level rise in the future that was based on tide gauge stations in Darwin and Milner Bay in Australia's Northern Territory. For this study, the investigated data included mean sea level data on BOM associated with the time period between 1990 and 2022. The study modeled data using four artificial intelligence techniques: SVR, AdaBoost, MLP, and CNN-BiGRU were the best performing algorithms in descending order. The MSL interpretation revealed that the same was 6 on an upward trend showing that the increase in materialism was influencing the consumers. 1 ± 1. It was considered that one of the weaknesses of this research was the fact that there were limited numbers of tide gauge locations in the Northern territory and therefore results and predictive models might not have been applicable in other climatic regions with have different oceanographic features.

Existing studies on sea level rise (SLR) prediction incorporate a combination of hybrid and machine learning algorithms with different strengths and weaknesses. The sea surface temperature and sea level height are increasing, as per a study on Gothenburg; however, its limitation lies in its reliance on historic data, which may cause underestimation in the future. While accuracy is impaired by the lack of high-frequency data and key parameters, LSTM models have proven to be more accurate. SVM and KNN performed better than traditional models in the Bosphorus Strait, but only within a limited geographic area. The SSA-LSTM models of the Yellow Sea are very accurate, but they are not generalizable beyond the region in question. Similarly, local environmental considerations are highlighted in BiLSTM applications for small island nations. Though constrained by sparse tidal gauge data, a new model in northern Australia that applied data decomposition and AI methods gave promising results. For more accurate and scalable predictions of SLR, these papers as a group point to the need for bigger, regionally adaptive models considering a range of environmental variables.

III. Problem Statement

Accurate forecasting of sea level rise is vital for effective coastal management; however, traditional approaches often rely heavily on historical data [22], overlooking future uncertainties and critical climatic or socio-economic factors [23]. To overcome these constraints, this research proposes a new method that incorporates T-GCN, attention mechanisms, and game theory-based optimization. T-GCNs are well suited to capture the intricate spatiotemporal dependencies inherent in sea level observations, while the attention mechanism chooses important time periods and spatial coordinates. Game theory optimization further improves model performance by imitating strategic interactions among important variables, tuning predictive results. This combined framework significantly enhances prediction accuracy, providing a robust and reliable instrument for backing long-term coastal resilience and adaptation planning.

IV. Methodology for Sea Level Rise Prediction Using Hybrid Methods

The interaction of sea levels with other parameters can be effectively predicted through the proposed methodology, which combines T-GCN, attention mechanisms, and game theory optimization. The process starts with the data where data from Kaggle of sea level change is then pre-processed by cleaning, normalizing from it and then subdividing it. Temporal and spatial features are reconstructed using a T-GCN model to

analyse the given data. To improve the prediction the attention mechanisms are used to increase the model's focus on the significant features. In federated learning approach, multiple clients participate and perform updates on the global model without accessing raw data. Game theory optimization extends the model by incorporating multiple objectives such as error frequencies and model overfitting. The above steps are then repeated several times till pre-specified convergence criteria are met and the final optimized model for predicting sea level rise is arrived. The proposed methodology is illustrated in Fig. 1.

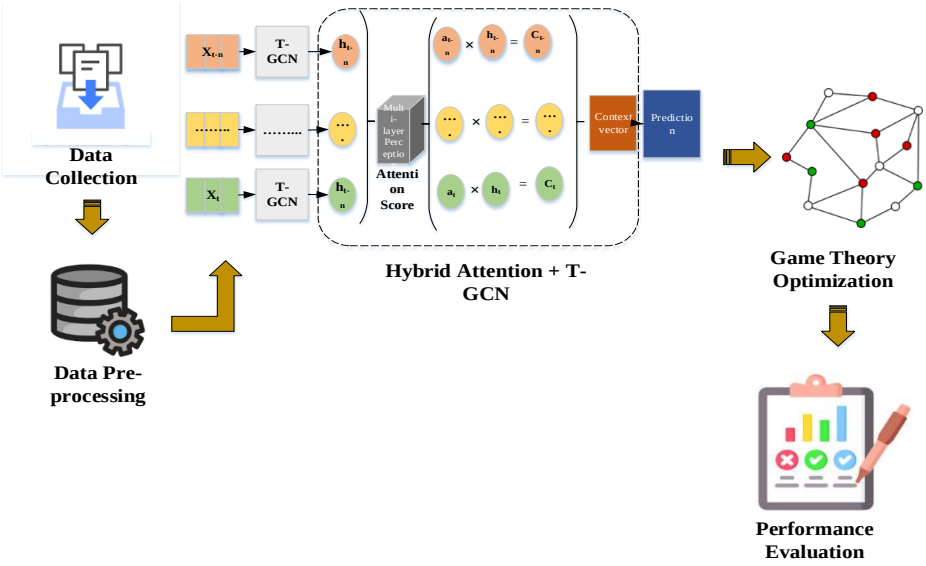


Fig. 1. Proposed methodology for sea level rise prediction.

A. Data Collection

The "Sea Level Change" Kaggle dataset [29], containing complete observations of sea level across a variety of geographic locations and historical time frames, is utilized in this study. Dates of measurement, sea level elevation, geolocation coordinates (latitude and longitude), and other climatic parameters such as temperature and precipitation are all present in the dataset. Sea level change has a huge coverage in terms of time, and thus is easier to understand long term trends and patterns in it. This research is seeking to improve the quality of predictions of sea level change with the data provided by attention and T-GCNs.

B. Data Pre-processing

The pre-processing ensure the data quality and useability in order to predict sea level rise using either mean or KNN imputation for missing values and outlier detection processes including the Z-score method to remove outliers. The next stage is normalizing the values of features using either a Min-Max Scale or Z-Score Normalization. Finally, temporal and geographical adjustments are used to synchronize data between uniform time intervals and uniform spatial resolution. These techniques are a part of the pre-processing procedure to ensure that the dataset is conducted properly for features to be extracted, and for predictive modeling.

1) Handling missing values: Handling missing values is an important step in data preprocessing to guarantee data trustworthiness. KNN addresses this problem by estimating the missing values with a weighted average of k nearest instances, based mainly on Euclidean distance. Weights are inversely proportional to distances, assuming that nearby data points have similar values. This method preserves size of data and natural data rhythms for providing quality assured data for an future analysis and model, as shown in Eq. (1) [24].

$$x_i = \frac{1}{k}\sum_{j=1}^{k} x_j \quad (1)$$

Where x_i are the values of k-nearest neighbors.

2) Data normalization: Min-Max normalization is a common transformation of calculations that make data range into a defined interval, most commonly [0,1]. This technique normalizes features so that the values lie between 0 and 1, while maintaining the relative position of the features data to each other and maximizing the importance of all features as they relate to the model characteristics. The normalization process is conducted by Eq. (2), [25]

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (2)$$

where the X is the value, X_{min} is the minimum value of a feature, whereas X_{max} is the maximum value of the feature. This transformation is most helpful in scenarios where the scale of features should be made consistent to enhance models' convergence as well as the performance of algorithms in the case of machine learning. This normalization becomes more important in a situation where the features have different units of measurement because this brings all of them to the same level so that the distance calculations and the training of the model will be favoured by all of the features. This can enhance the stability of the model and also increase its general performance by mitigating effects that come with the presence of features that are big and small in magnitude.

3) Temporal and spatial alignment: Synchronization: Data points are more reliable when they are obtained in harmonized

time intervals and spatial resolutions to avoid difference in results. This entails extrapolation of data, whereby the data is either interpolated or aggregated to correspond with the required frequency and location. For temporal alignment, interpolation which can be employed to approximate values in missing time points can be used. This is demonstrated by the following empirical Eq. (3), [26]

$$X_t = X_{t-1} + \frac{X_{t+1} - X_{t-1}}{2} \quad (3)$$

Where X_t represents the interpolated value at time t, X_{t-1} is the value at the previous time point, X_{t+1} and is the value at the subsequent time point.

C. T-GCN and Attention Mechanism for Feature Extraction and Classification

Sea level rise predictions need models capable of spatial dependencies between the coastal tracking stations and time series dynamics of the ocean processes. The current methods have certain limitations: (CNNs) are good in detecting spatial-based features but they lack the capability to incorporate sequential patterns, whereas (LSTMs) can capture temporal dependence patterns but not background spatial features. (GCNs) are strong at representational modeling of spatial structures but do not provide temporal modeling of evolving relationships and standalone attention mechanisms tend to over-fit due to disproportionate weighting of features. Such weaknesses make the traditional methods inadequate in capturing the highly nonlinear, and highly coupled nature of prediction of sea level rise. To overcome these shortcomings, the hybrid framework proposed has temporal GCN to learn spatiotemporal representations, attention mechanism to prioritize different features adaptively, and a game theory-based optimization strategy to balance tradeoffs between different objectives, hence enhancing robustness, accuracy, and generalizability in the coastal regions presented by different coastal environments.

The method of predicting sea level rise involves using dataset which contains sea surface temperature, wind speed, precipitation rates, and height. Then this data is pre-processed, where all the data is cleaned, normalized, and divided into train, validation and test sets. T-GCNs are then used to learn the spatial temporal structure and important features are then obtained. These features are then further brought through the use of attention mechanisms that aim to attend to important temporal and spatial details. Finally, the applicability of the hybrid model is evaluated in terms of RMSE, MAE and NSE with different level cross validation. The outcome that is expected from this approach is accuracy, completeness and flexibility of the numerical forecast of sea level increases.

1) T-GCN Model: In this section, the general structure of the algorithm is described as well as specifics of the actual algorithm utilized. For the purpose of modelling both spatial and temporal dependencies the model is used. The proposed model fuses the GRU and GCN. To model complicated geographical dependence, GCN is employed to capture the dependence of radar networks' topological structure and temporal dependence GRU is used to learn the dynamic evolution of radar networks. The multi-radar network's objective in this study is to forecast the radar feature at a given time utilizing network detection data. Any of the following can be the radar characteristic in this method: beam shaping, bandwidth, amplitude, centre frequency, or sampling rate.

a) Input layer: The input layer receives multi-different time chain data collected from the network of geographically distributed C-level monitoring stations. Each data example includes several parameters, such as sea level height, sea surface temperature, atmospheric pressure, wind speed and rainfall. These comments are structured in a temporary sequence and are locally connected, which means that each station is considered a node in a graph. The edges between the nodes represent spatial relationships – either geographical proximity or statistically learnt correlation (e.g., Pearson correlation or mutual information). This graph structure allows the model to understand how the sea level in one place can change or affect the neighbouring areas. The time series aspect ensures that the temporary development of these characteristics is preserved and can be effectively modelled by downstream components such as GCN and GRU layers.

b) GCN block: The graph is responsible for learning spatial dependence between sea level monitoring stations by taking advantage of the graph structure of block data. GCN block models spatial dependence between sea level monitoring stations. It is operated by collecting information about the facility from neighbours of each node based on the adjacent matrix A. Layer is defined as a graph conversion operation on l+1:[27]

$$H^{(l+1)} = \sigma(\tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2}H^{(l)}W^{(l)}) \quad (4)$$

In Eq. (4), the adjacency matrix with added self-loops were denoted as $\tilde{A} = A + 1$. The degree of matrix of $\tilde{A}$ is denoted as $\tilde{D}$. The input feature matrix at layer l is denoted as $H^{(l)}$ and the trainable weighted matrix is denoted as $W^{(l)}$ and finally the activation function is denoted as σ. Using an adjacency matrix, which defines the relationship between spatial nodes (stations), GCN collects information from immediate neighbours of each node. Through the resolution operation conducted on the graph, this block learns spatial representations by smoothing or mixing information in the nodes nearby, which is important to understand the regional dependence. For example, an increase in sea level at a station may indicate or affect the same pattern in nearby areas. By encoding this relevant spatial information, the GCN converts the feature vector of each node into a rich representation that captures how the local sea level behaviour is shaped by its spatial reference.

c) GRU block: The GRU block captures the temporary dependence of data by processing the sequence of spatial characteristics produced by GCN over time. GRUs are a type of RNN that are designed to handle long-term dependence in sequential data without suffering from missing gradients. They work using two gates: the update gate (z_t) determines how much previous information is retained, and the reset gate (r_t) controls how much time of the past is forgotten at each step.

Update Gate [28]

$$z_t = \sigma(W_z X_t + U_z h_{t-1} + b_z) \quad (5)$$

Reset Gate

$$r_t = \sigma(W_r X_t + U_r h_{t-1} + b_r) \quad (6)$$

These help the gate models to maintain a balance between remembering important historical trends and adapting to recent changes in data. By recurring through the stages of time, GRU learns sudden changes in temporary diversity, trends and sea level patterns. This can recall the model that simply models the model with a temporary awareness to predict future sea level values.

d) Hidden output: The final output of the GRU block is Hidden State, which enforces both spatial and temporary information learnt from data. This hidden position acts as a compact representation of a multi-compact time chain, briefly presenting the pattern seen in space (via GCN) and time (through GRU). This includes the required future features, such as the impact of neighbouring stations, historical sea level trends and the importance of different time intervals. The hidden state can be passed in later layers – such as a mediation mechanism, which refines further convenience, or directly predicts an increase in future sea levels for a prediction head (e.g., a dense layer or MLP). Its compact yet expressive nature ensures efficient modelling while maintaining all the necessary information for accurate and strong prediction.

Fig. 2 illustrates the T-GCN, which captures spatial-temporal relationships in graph-structured time series data by integrating GRU and GCN. GRUs employ hidden states h_t, regulated by reset R_t and update Z_t gates, to learn temporal patterns, whereas GCNs utilize spatial information from input nodes (X1–X4) to generate outputs (Z1–Z4). Due to its capacity to process dynamic spatial-temporal information, T-GCN is perfectly suited for sea level rise prediction with enhanced accuracy of forecasts.

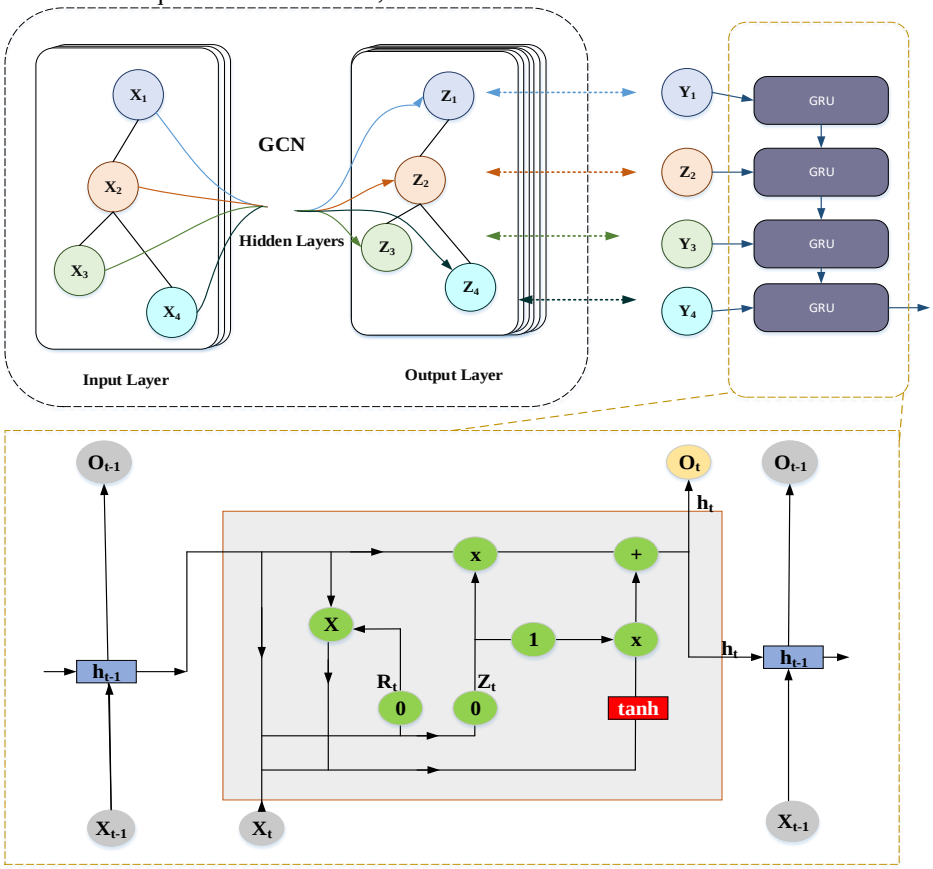


Fig. 2. Architecture of T-GCN.

2) Integration of hybrid temporal graph convolutional networks and attention mechanism: The proposed architecture enhances sea level rise prediction by integrating T-GCN with an attention mechanism. By representing geographic locations as graph nodes and employing graph convolutions to learn patterns over time, T-GCNs effectively simulate spatiotemporal dependencies. An attention mechanism that employs query, key, and value computations to assign higher weights to the most relevant spatial and temporal features is then applied to fine-tune the retrieved features. To emphasize critical information while suppressing less informative parts, attention scores are scaled by a SoftMax algorithm. As the final prediction layer runs over the improved features, the risk analysis for coastal planning is made more robust and accurate. The hybrid architecture is given in Fig. 3.

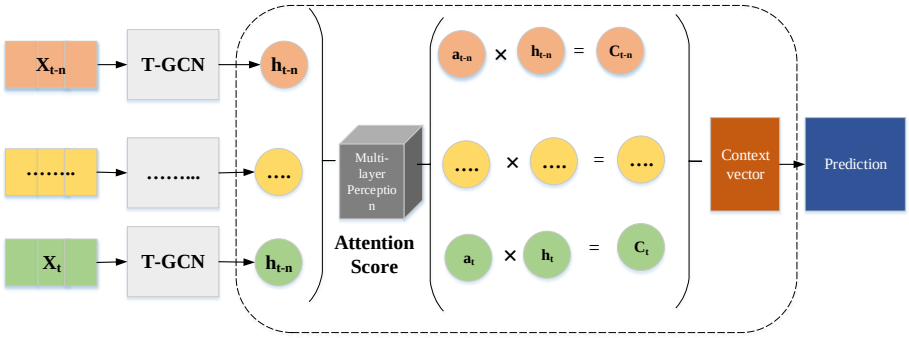


Fig. 3. Architecture of hybrid attention based T-GCN.

Fig. 3 depicts a hybrid model integrating T-GCN with attention for precise time-series prediction. Sequential input sequences from X_{t-n} to X_t are fed into T-GCN modules to learn spatial and temporal relationships, producing hidden representations. These are fed into a multilayer perceptron and an attention mechanism that places dynamic weights on every time step. A context vector is calculated by summing attention-weighted hidden states, focusing on pertinent temporal features. This context vector is utilized for prediction, allowing the model to provide solid and interpretable predictions for intricate spatiotemporal phenomena such as sea level rise.

Algorithm 1: Algorithm for Sea Level Rise Prediction Using T-GCN, Attention Mechanisms, and Game Theory Optimization

```
Begin
  Input: Dataset D, adjacency matrix A, time steps T, iterations K
  For each missing value in D
    Apply KNN imputation
  End
  For each feature in D
    Compute Z-score
    If |Z-score| > threshold Then
      Remove outlier
    End
  End
  Normalize features and align data temporally and spatially
  Define nodes V and construct adjacency matrix A
  For each time step t in T
    Build temporal graph Gt
    Apply GCN and GRU to extract spatiotemporal features
  End
  For each extracted feature
    Compute attention score
    If score > threshold Then
      Assign higher weight
    Else
      Assign lower weight
    End
  End
  Initialize k = 1
  While k ⩽ K
    Define objectives and update parameters
    Balance trade-offs using game theory
    Increment k
  End
  Repeat
    Extract features, apply attention, optimize
  Until stopping criteria satisfied
  Return predicted sea level values
End
  Output: Predicted sea level values
```

V. Results and Discussion

The research proves the effectiveness of the proposed T-GCN augmented with attention mechanisms and optimized using game theory approaches, leading to enhanced performance for sea level rise prediction compared to current models. The predicted values exhibit excellent correspondence to the actual measured values, which indicates the ability of the model to successfully extract prior spatial and temporal influences from the data. Moreover, the model reached a very low MAE for values, surpassing equivalent low prediction uncertainty. This serves to further demonstrate the advantage of combining temporal graph-based features with attention mechanisms to effectively incorporate complex spatiotemporal dependencies often inherent in sea level data sets.

A. Historical Trend of Global Average Sea Level Change Over Time

Fig. 4 shows the global, temporal evidence of sea level elevation, represented in millimeters (y-axis) over a time series (x-axis). The figure demonstrates a remarkable upward progression that indicates that sea levels continue to rise, at an ever-increasing rate. For the measured time frame, the sea level has risen by about 200 millimeters (about 8 inches), which points toward serious environmental issues. This increase has dire implications for coastal regions, such as saltwater intrusion into freshwater systems, higher frequency of flooding, more rapid coastal erosion, and population displacement near shorelines.

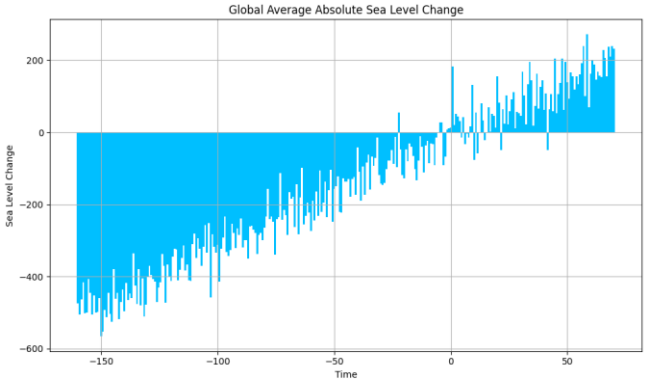


Fig. 4. Historical trend of global average sea level change over time.

B. Rolling Mean and Rolling Deviation of Sea Level Change Over Time

Fig. 5 presents a time series plot of the rolling mean and deviation, illustrating the overall trend and fluctuations in sea level change. The rolling mean shows an initial sea level decline from -200, gradually rising, crossing the x-axis at time 0, and reaching 200—indicating a clear upward trend over the period. Thus, while the mean shows a steady upward trend over the period, the rolling deviation remains relatively stable, indicating consistent variability around the trend. Sea level rise occurs as a continuous process, but the fluctuations around its rate remain relatively stable over time, indicating no consistent pattern of acceleration or deceleration in the short term.

C. Training and Testing Accuracy

Fig. 6 shows the accuracy curves for only the training and the testing phase of the improved Temporal Graph. Convolutional Network known as T-GCN over 100 epochs. Here the solid blue line stands for the training accuracy and the orange dotted line stands for the testing accuracy. At first, both increase significantly; training one rises above 100 % while testing one increases to nearly 90 % by about the 10th iteration. This means that the model gains a relatively fast adjustment to the patterns and relationships of the data. After this point, both the accuracies decrease a little and then become almost stagnant with the training accuracy being slightly higher than testing accuracy. Both accuracies for train and validate data are quite impressive and gradually rises until the 100th epoch, and are very close to the 100% and show model's capability of generalizing well though unseen data after number of epochs.

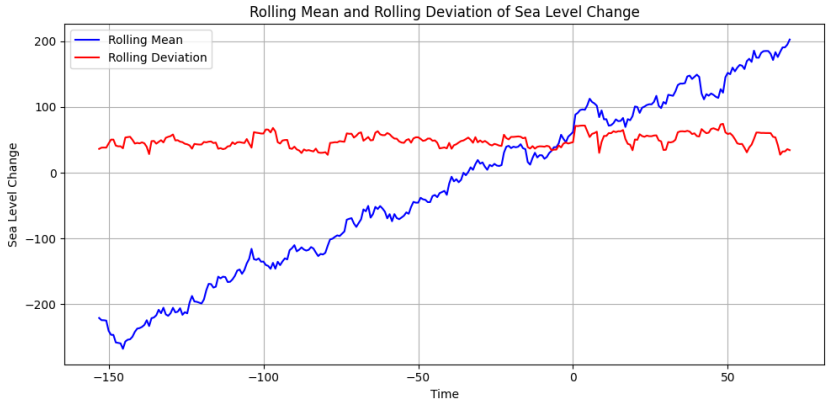


Fig. 5. Rolling mean and rolling deviation of sea level change over time.

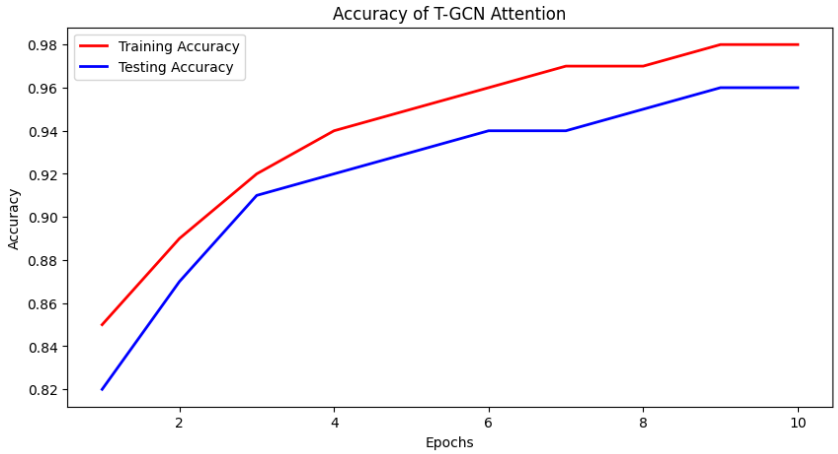


Fig. 6. Training and testing accuracy.

D. Training and Testing Loss

Fig. 7 shows the training and testing loss curves of T-GCN model with attention mechanisms on 60 epochs. The blue curve represents the training loss whereas, the orange line indicates the testing loss. First, both losses are high; the training loss is more than two for all the epochs except epoch 15. 5 and the testing loss is slightly less than 2. 5, which means great discrepancies in the predictions during the initial phases of the model training process. However, as the training process goes further both losses rapidly decrease, the training loss reaches even the negative values below 0. 5 in the 10 th epoch and the testing loss which also trended downward in the same way. As for epoch 30, both the training and testing losses reduce continuously and smoothly which almost reached the testing value equal to zero after 60 epochs. The training and testing loss curves of T-GCN model with attention mechanisms on 60 epochs.

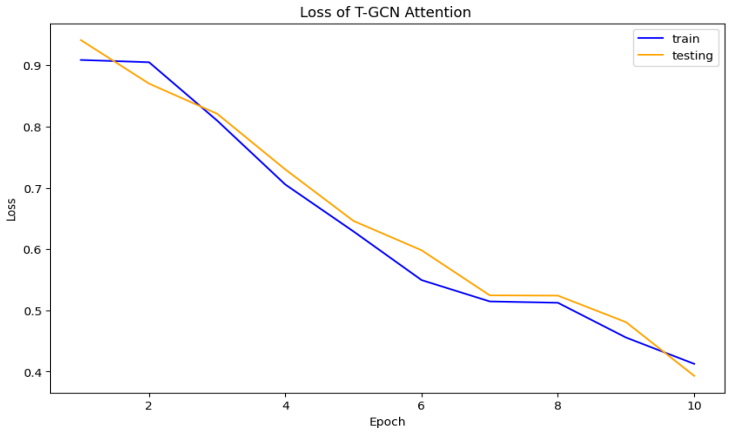


Fig. 7. Training and testing loss.

E. Performance Evaluation

Correlation coefficient, Willmott's Index of Agreement, Nash-Sutcliffe Coefficient, Legates and McCabe's Index, and Nash-Sutcliffe Coefficient are among the performance assessment metrics used to assess the models' dependability in terms of predicted accuracy. In order to determine the extent of the model's inaccuracy and to compare the outcomes, other error statistics such as Root Mean Squared inaccuracy, Mean Absolute Error, Relative Root Mean Squared Error, and Mean Absolute Percentage Error are also utilized.

1) Correlation Coefficient (r): Coefficient of determination reflects the reliability of the regression equation and the correlation coefficient r quantifies the linearity of the observed and simulated data. It varies between -1 and + 1 where a value closer to - 1 represents a perfect negative correlation and a value closer to + 1 perfect positive correlation, no linear relationship is given by Eq. (7) as follows, [29].

$$r = \frac{\sum_{i=1}^{n} DO_i - MDODS_i - MDS}{\sum_{i=1}^{n} DO_i - MDO^2 \sum_{i=1}^{n} DS_i - MDS^2} \quad (7)$$

2) Willmott's index of agreement: Willmott's Index of Agreement (d) is a widely used measure of prediction accuracy against a group of consensus observed data. The expected range of values is from 0 to 1, with 0 indicating no agreement at all, and 1 indicating perfect agreement between predicted and observed values. The index differs from basic correlation measures in that it emphasizes errors by their magnitude, providing a more sensitive measure of prediction accuracy.

$$d = 1 - \frac{\sum_{i=1}^{n} DO_i - DS_i^2}{\sum_{i=1}^{n} DS_i - MDO + DO_i - MDS^2} \quad (8)$$

3) Nash-sutcliffe coefficient: Consequently, to check the potential accuracy of fitted models, the Nash-Sutcliffe Coefficient NS is used. It also shows that an NS value of 1 reflects perfect model performance while an NSNSNS of 0 reflects the model performance in terms of observed means. This is given by Eq. (9),

$$NS = 1 - \frac{\sum_{i=1}^{n} DO_i - DS_i^2}{\sum_{i=1}^{n} DO_i - MDO^2}, -\infty \leq NS \leq 1 \quad (9)$$

4) Legates and McCabe's index: LM is the other measure that is used to evaluate the performance of given models. It highlights the gross dissimilarities of observed and simulated data, which is downplayed for outlying values. LM is a value between 0 and 1 with the higher value representing better results of the model. This is provided by Eq. (10)

$$LM = 1 - \frac{\sum_{i=1}^{n} |DS_i - DO_i|}{\sum_{i=1}^{n} |DO_i - \overline{DO}|} \quad (10)$$

5) Root mean square error: RMSE is a measure of the ordinary used in the assessment of the differences between the estimations and the observations. It imposes more penalty for larger errors than for small ones, owing to the fact that, the residuals are squared before averaging. This corresponds to the value expressed by the Eq. (11),

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} DS_i - DO_i^2} \quad (11)$$

6) Mean Absolute Error: MAE is computed from the average of the absolute differences between the predicted and the actual values, without a regard to the sign of the differences. Basically, it is easy to understand and easy to interpret. This is as given by the Eq. (12).

$$MAE = \frac{1}{n}\sum_{i=1}^{n} DS_i - DO_i^2 \quad (12)$$

7) Relative Root Mean Square Error: RRMSE gives an RMSE percentile value which is the RMSE of a normalized set of values, expressed as a percentage. It scales the RMSE to the range of the observed data thus being suitable when comparing models on different ranges of data. This is given by Eq. (13).

$$RRMSE = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n} DS_i - DO_i^2}}{\sqrt{\frac{1}{n}\sum_{i=1}^{n} DO_i}} \times 100 \quad (13)$$

8) Mean Absolute Percentage Error: MAPE expresses the average prediction error as a percentage of the observed values. It's useful for understanding the relative size of errors. This is given by Eq. (14),

$$MAPE = \frac{1}{n}\sum_{n}^{i=1}\frac{DS_i - DO_i}{DO_i} \times 100 \quad (14)$$

Where DO_i_i is the observed data, DS_i is the simulated data, MDO is the mean of observed data, and MDS is the mean of simulated data.

Table I and Fig. 8 offer a summary of the performance of the T-GCN + Attention model. The model is good at prediction and has close correspondence between predicted and observed values as indicated by the Nash-Sutcliffe Efficiency of 0.9925, Willmott's Index of Agreement of 0.9958, and Correlation Coefficient of 0.9965. The MAE and RMSE, which reflect small prediction errors, are low at 0.0322 and 0.0412, respectively, yet the Legates and McCabe Index is comparatively high at 0.9321.Though the model is still very reliable across evaluation metrics, relatively high Mean Absolute Percentage Error (1.95%) and Relative RMSE (2.38%) values reflect high unpredictability.

TABLE I. PERFORMANCE AND ERROR METRICS

Metric	T-GCN + Attention
Nash-Sutcliffe Coefficient	0.992452
Willmott's Index of Agreement	0.995804
Correlation Coefficient	0.996512
Legates and McCabe Index	0.932145
RMSE	0.041203
MAE	0.032154
RRMSE	2.381204
MAPE	1.953214

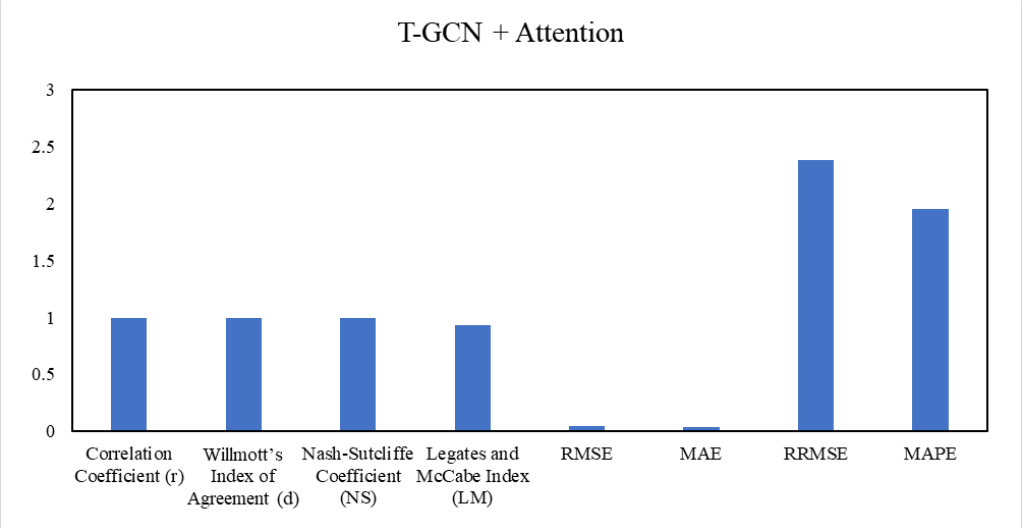


Fig. 8. Performance and error metrics comparison.

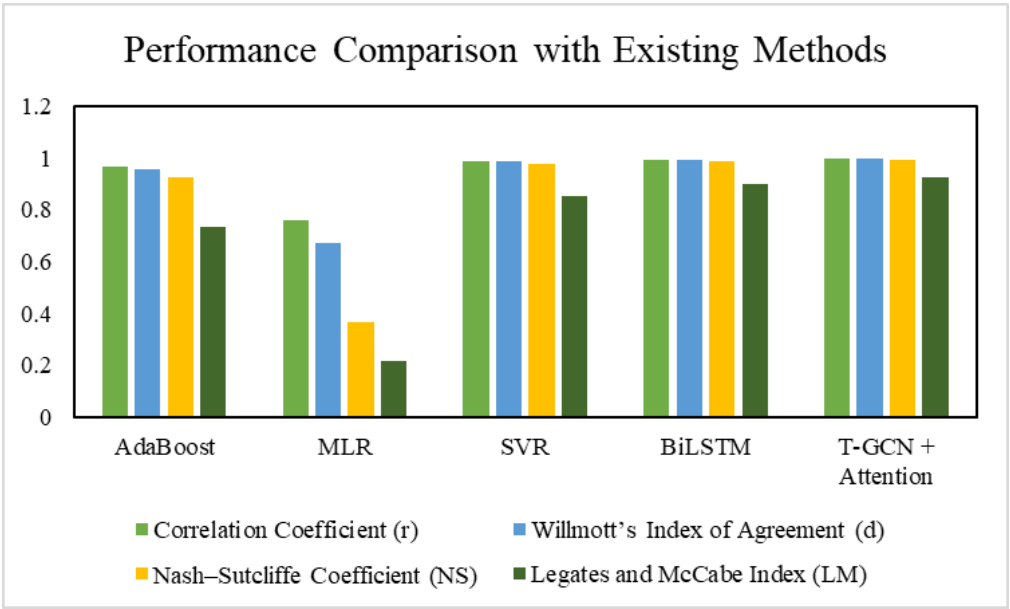


Fig. 9. Performance comparison with existing methods.

Fig. 9 and Table II illustrate a comparative study of five models—AdaBoost, MLR, SVR, BiLSTM, and T-GCN with Attention—based on four performance metrics: Correlation Coefficient (r), Willmott's Index of Agreement (d), Nash–Sutcliffe Efficiency (NS), and the Legates and McCabe Index (LM). The Correlation Coefficient (r) varies between 0.759946 for MLR and 0.996502 for the T-GCN + Attention model, which shows that the latter has the highest linear correlation between predicted and observed values. The Nash–Sutcliffe Efficiency (NS), which is one of the most important indicators of predictive performance, varies from 0.368981 for MLR to 0.993021 for T-GCN + Attention, which indicates greater forecast capacity for the proposed hybrid model. Also, the Legates and McCabe Index (LM), the measure of reliability of model predictions, varies from 0.217703 for MLR to 0.925741 for T-GCN + Attention. Overall, the results clearly show that the proposed T-GCN + Attention model outperforms the other methods on all evaluation metrics, and it confirms there are overall strong, accurate, and effective in predicting sea level rise.

TABLE II. PERFORMANCE COMPARISON WITH EXISTING METHODS

Model	Correlation Coefficient	Willmott's Index of Agreement	Nash–Sutcliffe Coefficient	Legates and McCabe Index
CNN-BiGRU [30]	0.988909	0.987155	0.974866	0.952321
BiLSTM [31]	0.974207	0.964079	0.988219	0.939868
SVMD-BiLSTM [32]	0.954311	0.987647	0.983809	0.913546
CEEMDAN-CNN[33]	0.968946	0.982739	0.988981	0.927703
Proposed T-GCN + Attention	0.996502	0.996312	0.993021	0.985741

a) Paired t-test on MAE between Proposed Model and BiLSTM

TABLE III. PAIRED TWO-TAILED T-TEST

Fold	T-GCN + Attention (Proposed)	BiLSTM
1	0.031	0.043
2	0.032	0.046
3	0.033	0.042
4	0.030	0.045
5	0.031	0.044

Table III paired two-tailed T-test was held in five cross-validation folds to statistically evaluate the performance difference in the meaning of the T-GCN + meditation model and the Bi-LSTM baseline in the proposed T-GCN + MAE. The results detected a low MAE for the proposed model in all folds, and the calculated T-statistic was much higher than the significant value, indicating sufficient difference in performance. With a particularly low p-value ($p < 0.01$), the improvement obtained by the proposed hybrid model is statistically important. This confirms that the T-GCN + attention model not only provides a better future accuracy, but continuously does so, offering a strong and reliable solution to increase sea level compared to traditional deep learning approaches.

```
Paired t-test result:
t-statistic: 13.4164
p-value: 0.000195
Result is statistically significant (p < 0.05).
```

Fig. 10. T-test outcome.

To validate the performance of the T-GCN + attention model proposed on baseline methods, a coupled two-wheel T-test on MAE values obtained from 5-fold cross-fold was organised as mentioned in Fig. 10. Results ($T = 13.4$, $P < 0.01$) indicate a statistically significant decrease in predicted error compared to BILSTM; it confirms that the proposed model provides a consistent and average improvement.

F. Discussion

Precise forecast of sea level rise is essential for climate mitigation and adaptation planning. A new hybrid model that combines T-GCN, attention mechanisms, and game theory optimization is introduced in this research to overcome the limitations of classical models to address highly complex spatiotemporal patterns. The T-GCN models temporal dependencies and spatial correlations, and the attention mechanism emphasizes prominent features such as glacier melt and ocean thermal expansion. Game theory reinforces model strength by encoding feature interactions as a multi-objective strategy, enhancing generalizability and mitigating overfitting. Results of evaluation indicate high predictive efficacy and minimal error rates. The framework is a solid instrument for scientists and policymakers that facilitates data-driven, fine-grained planning for coastal resilience to climate change.

VI. CONCLUSION AND FUTURE WORK

The study introduced a novel Hybrid Forecasting Framework that integrates T-GCN, attention mechanisms and game theory-based adaptations to increase the accuracy of sea level rise prediction. Mixing GCN's spatial modelling abilities with the GRU's temporary learning strength, the proposed T-GCN structure effectively captures the complex spatiotemporal dependence contained in sea level figures. The meditation mechanism further enhances performance by emphasising important characteristics in both dimensions, while the game theory refines the model by reducing optimisation over fitting and ensuring balanced performance in different environmental conditions. As displayed through a high correlation coefficient and low error rates, the model displays strong future-stating capacity, making it a valuable.

Further, the model can be increased by incorporating additional data sources such as demographic profiles, population density, regional melted water rates and extreme weather indicators. These enrichers will not only improve future accuracy but will also enable field-specific risk assessment. Real-time data acquisition and integrating adaptive local calibration will make the system more dynamic and responsible for on-ground changes. Ultimately, deployment of this model within the decision-support structure can assist policy makers and planners in crafting data-operated responses for climate-inspired sea level changes, promoting permanent and flexible coastal growth.

REFERENCES

[1] H. Nazarnia, M. Nazarnia, H. Sarmasti, and W. O. Wills, "A systematic review of civil and environmental infrastructures for coastal adaptation to sea level rise," Civ. Eng. J., vol. 6, no. 7, pp. 1375–1399, 2020.

[2] J. Singh and others, "The Accelerating Melting of Glaciers: Impacts on Earth and Emerging Challenges," Int. J. Adv. Res. Interdiscip. Sci. Endeav., vol. 2, no. 1, pp. 401–414, 2025.

[3] P. Chahal, "Global warming and greenhouse effect," Conserv. Sustain. Environ. Justice India, vol. 95, 2020.

[4] J. O. Kayode-Edwards, I. I. Kayode-Edwards, and D. O. Kayode-Edwards, "Climate change in the Arctic," in Arctic Marine Ecotoxicology: Climate Change, Pollutants, and Their Far-Reaching Effects, Springer, 2024, pp. 71–96.

[5] G. Griggs and B. G. Reguero, "Coastal adaptation to climate change and sea-level rise," Water, vol. 13, no. 16, p. 2151, 2021.

[6] R. K. Upadhyay and others, "Markers for global climate change and its impact on social, biological and ecological systems: A review," Am. J. Clim. Change, vol. 9, no. 03, p. 159, 2020.

[7] O. Ruth, D. Lagat, and O. Lilian, "Linking adaptation and mitigation toward a resilient and robust infrastructure sector in Kenya," Afr. Handb. Clim. Change Adapt., pp. 1–19, 2020.

[8] M. S. Islam, "Sea-Level Changes Along Bangladesh Coast: How Much Do We Know About It?," in Applied geomorphology and contemporary Issues, Springer, 2022, pp. 611–633.

[9] J. R. Bernhardt, M. I. O'Connor, J. M. Sunday, and A. Gonzalez, "Life in fluctuating environments," Philos. Trans. R. Soc. B, vol. 375, no. 1814, p. 20190454, 2020.

[10] T. Harvey et al., "Ocean mass, sterodynamic effects, and vertical land motion largely explain US coast relative sea level rise," Commun. Earth Environ., vol. 2, no. 1, p. 233, 2021.

[11] A. Bukvic, G. Rohat, A. Apotsos, and A. de Sherbinin, "A systematic review of coastal vulnerability mapping," Sustainability, vol. 12, no. 7, p. 2822, 2020.

[12] A. Tsatsaris et al., "Geoinformation technologies in support of environmental hazards monitoring under climate change: An extensive review," ISPRS Int. J. Geo-Inf., vol. 10, no. 2, p. 94, 2021.

[13] C. Jörges, C. Berkenbrink, H. Gottschalk, and B. Stumpe, "Spatial ocean wave height prediction with CNN mixed-data deep neural networks using random field simulated bathymetry," Ocean Eng., vol. 271, p. 113699, 2023.

[14] F. Noor et al., "Water level forecasting using spatiotemporal attention-based long short-term memory network," Water, vol. 14, no. 4, p. 612, 2022.

[15] J. Ma, L. Yao, L. Deng, Q. Yang, Y. Chen, and C. Ouyang, "Spatiotemporal prediction of landslide deformation based on temporal graph convolutional networks," 2024.

[16] O. Memarian Sorkhabi, B. Shadmanfar, and M. M. Al-Amidi, "Deep learning of sea-level variability and flood for coastal city resilience," City Environ. Interact., vol. 17, p. 100098, Jan. 2023, doi: 10.1016/j.cacint.2022.100098.

[17] G. Accarino et al., "A multi-model architecture based on Long Short-Term Memory neural networks for multi-step sea level forecasting," Future Gener. Comput. Syst., vol. 124, pp. 1–9, Nov. 2021, doi: 10.1016/j.future.2021.05.008.

[18] A. Altunkaynak and E. Kartal, "Transfer sea level learning in the Bosphorus Strait by wavelet based machine learning methods," Ocean Eng., vol. 233, p. 109116, Aug. 2021, doi: 10.1016/j.oceaneng.2021.109116.

[19] [N. Raj, "Prediction of Sea Level with Vertical Land Movement Correction Using Deep Learning," Mathematics, vol. 10, no. 23, p. 4533, Nov. 2022, doi: 10.3390/math10234533.

[20] J. Zhao, R. Cai, and W. Sun, "Regional sea level changes prediction integrated with singular spectrum analysis and long-short-term memory network," Adv. Space Res., vol. 68, no. 11, pp. 4534–4543, Dec. 2021, doi: 10.1016/j.asr.2021.08.017.

[21] N. Raj, J. Murali, L. Singh-Peterson, and N. Downs, "Prediction of Sea Level Using Double Data Decomposition and Hybrid Deep Learning Model for Northern Territory, Australia," Mathematics, vol. 12, no. 15, p. 2376, Jul. 2024, doi: 10.3390/math12152376.

[22] Y. Qin, C. Su, D. Chu, J. Zhang, and J. Song, "A review of application of machine learning in storm surge problems," J. Mar. Sci. Eng., vol. 11, no. 9, p. 1729, 2023.

[23] N. A. A. B. S. Bahari et al., "Predicting sea level rise using artificial intelligence: a review," Arch. Comput. Methods Eng., vol. 30, no. 7, pp. 4045–4062, 2023.

[24] B. P. Duarte, A. C. Atkinson, and N. Oliveira, "Compound optimal design of experiments–Semidefinite Programming formulations," Optim. Eng., pp. 1–23, 2025.

[25] M. Prasad and T. Srikanth, "Clustering Accuracy Improvement Using Modified Min-Max Normalization Technique," 2024.

[26] B. Buonaguidi, "Numerical approaches to finite-horizon optimal stopping problems for the Shiryaev process," Stat. Comput., vol. 35, no. 6, pp. 1–22, 2025.

[27] Z. Chen, L. Fu, J. Yao, W. Guo, C. Plant, and S. Wang, "Learnable graph convolutional network and feature fusion for multi-view learning," Inf. Fusion, vol. 95, pp. 109–119, 2023.

[28] Y. Tian, H. Cao, D. Yan, J. Chen, and Y. Hua, "Spatiotemporal pattern evolution and quantitative prediction of electrical carbon emissions from a demand-side perspective in urban areas," Sci. Rep., vol. 15, no. 1, p. 25097, 2025.

[29] N. Raj, J. Murali, L. Singh-Peterson, and N. Downs, "Prediction of Sea Level Using Double Data Decomposition and Hybrid Deep Learning Model for Northern Territory, Australia," Mathematics, vol. 12, no. 15, p. 2376, 2024.

[30] N. Raj, J. Murali, L. Singh-Peterson, and N. Downs, "Prediction of Sea Level Using Double Data Decomposition and Hybrid Deep Learning Model for Northern Territory, Australia," Mathematics, vol. 12, no. 15, p. 2376, 2024.

[31] N. Raj, "Prediction of sea level with vertical land movement correction using deep learning," Mathematics, vol. 10, no. 23, p. 4533, 2022.

[32] N. Raj and J. Brown, "Prediction of mean sea level with GNSS-VLM correction using a hybrid deep learning model in Australia," Remote Sens., vol. 15, no. 11, p. 2881, 2023.

[33] N. Raj, Z. Gharineiat, A. A. M. Ahmed, and Y. Stepanyants, "Assessment and prediction of sea level trend in the South Pacific Region," Remote Sens., vol. 14, no. 4, p. 986, 2022.

Securing Image Messages Using Secure Hash Algorithm 3, Chaos Scheme, and DNA Encoding

Amer Sharif*, Dian Rachmawati, Wilbert
Dept. of Computer Science, Faculty of Computer Science and Information Technology,
Universitas Sumatera Utara, Medan, Indonesia

***Abstract*—Security is an essential aspect to consider during data transmission, especially images. Threats that may occur during image transmission include images being stolen by third parties. One way to secure images is through encryption-decryption processes using cryptographic algorithms. One of the algorithms developed for image security involves combining a chaos scheme, DNA encoding, and hashing. Chaos scheme refers to a system sensitive to initial conditions, resulting in behavior that is difficult to predict or appears random. DNA encoding is the process of converting bits into a DNA sequence. Hashing is a mathematical function which takes variable inputs and converts them into a binary sequence with fixed length. In this research, security enhancement is achieved by replacing the hashing algorithm with Secure Hash Algorithm (SHA) 3 Keccak. This study successfully implemented cryptographic algorithms into a website that can simulate image encryption-decryption processes in about 15 seconds per process. The effectiveness of the algorithm used has also been tested in abstracting images through Mean Squared Error (MSE) and Peak-Signal-to-Noise-Ratio (PSNR) evaluations. Obtained MSE values of 0 and PSNR values of infinity indicated that the original images and decrypted images are identical.**

***Keywords*—Image encryption; image decryption; chaos scheme; DNA encoding; secure hash algorithm 3 Keccak**

I. Introduction

The advancement of digital technology has led to an increased demand for data, especially images. Despite being a routine activity, the transmission of images between parties involves significant security risks, such as data theft by third parties. Therefore, it is crucial to develop methods for securing images during transmission.

Image encryption is an effective method for securing images. Encryption involves converting plaintext into ciphertext, making the data unreadable without a decryption key. This process ensures that only the intended recipient, who possesses the key, can convert the ciphertext back into readable plaintext, thus securing the image data during transmission.

This research explores the use of symmetric key cryptography, combining a chaos scheme and DNA encoding, alongside the Secure Hash Algorithm (SHA) 3 hashing algorithm, specifically Keccak. Commonly used hashing algorithms like SHA-1 and SHA-2 have known vulnerabilities, prompting the National Institute of Standards and Technology (NIST) to seek a more secure alternative, resulting in the adoption of the Keccak algorithm as the SHA-3 standard.

The chaos scheme generates semi-random numbers without repeating periods and is highly sensitive to initial parameter changes. This sensitivity ensures significant differences in output with minor changes in input, enhancing encryption robustness. Despite producing semi-random values, the chaos system is deterministic, meaning it can reproduce the same values with the same initial parameters, which is crucial for reliable encryption and decryption processes.

DNA encoding utilizes the properties of DNA sequences, where the bases A (Adenine) pair with T (Thymine) and C (Cytosine) pair with G (Guanine). By representing these DNA bases in a binary format, plaintext can be encoded into a secure format. This method, combined with the chaos scheme, has shown promise in previous studies, such as those by Belazi et al. [1], Wang and Su [15], and Zhong et al. [16].

Previous research indicates that the SHA-3 algorithm offers enhanced security compared to SHA-2 [10]. By integrating the chaos scheme, DNA encoding, and the SHA-3 algorithm, this study aims to develop a more secure image encryption method, potentially achieving higher security levels than existing techniques. The system developed may later be considered as an alternative means of image security in communication applications.

This study is organized next by enlisting previous studies in image encryption utilizing chaos and DNA algorithms and studies involving hash functions, followed by a literature review of the principal theoretical concepts. Following that is the problem analysis and system design, continued with implementation and system testing. Discussion and final conclusion with some considerations for future work are provided in the end.

II. Previous Studies

Several studies related to this approach have been conducted. Belazi et al. [1] obtained a cryptographic algorithm which met security criteria such as a large key space, resistance to statistical attacks, and a linear run time, enabling efficient execution. Alrubiae et al. [2] leveraged double-dynamic DNA sequence encryption and a chaotic 2D logistic map to produce a novel image encryption algorithm which resisted common attacks. Hussain et al. [4] developed substitution boxes by the use of chaotic logistic maps in linear fractional transformation. Arif et al. [5] used S-boxes, a hash

*Corresponding Author, Email: - amer.sharif@usu.ac.id

function, and chaos scheme to produce a secure cryptographic algorithm for images. Suryadi et al. [8] utilized a transposition scheme based on a chaos function, namely Arnold's cat map function, as a form of transposition resulting in a large key space with very low sensitivity. Munir [11] developed a simple image encryption system using a pseudo-one-time pad built upon a chaos function. Wang and Su [15] utilized the Piecewise Linear Chaotic Map (PWLCM) and Logistic Map to generate all parameters the encryption algorithm needed, combined with DNA encoding. Zhong et al. [16] combined chaos theory and a 2D Chebyshev-Sine map to achieve image encryption resistant to differential and selective plaintext attacks. Pittalia [10] compared the effectiveness of hash functions in cryptography, while Patil and Karule [7] specifically discussed the design and implementation of the Keccak hash function for cryptography.

III. Literature Review

A. Cryptography

Cryptography is a method for securing data using code algorithms, hashes, and signatures. It can protect data at rest, in transit, or in use. For example, cryptography can secure data stored on a hard drive, data being transferred between parties through electronic communication, or data being processed during calculations [12].

Cryptography has four main objectives: confidentiality, ensuring that information is accessible only to authorized parties; integrity, ensuring that information is not manipulated; authentication, confirming the authenticity of information and the identity of users; and non-repudiation, preventing users from denying their previous actions [14].

Cryptography generally is divided into two categories: symmetric cryptography, which utilize the same key for encryption, i.e. converting the readable plaintext/image into unreadable ciphertext/image, and decryption, i.e. converting the unreadable ciphertext/image back into the readable plaintext/image. On the other hand, asymmetric cryptography utilizes a set of different keys (called public and private keys) for encryption and decryption [14]. This research employs symmetric cryptography.

B. Secure Hash Algorithm

Secure Hash Algorithm (SHA) or often known as *one-way functions,* are mathematical functions which accept variable-length input and convert it into a binary string with a fixed length of 128 upto 512 bits. SHA are employed to ensure data integrity and authenticity [6] [14].

There are two criteria which serve as a standard for the security of an SHA algorithm:

1) Preimage resistance: Preimage resistance means for an arbitrary hash value, it is very difficult to obtain the input which produced that particular hash value.

2) Collision resistance: Collision resistance means it is very difficult to obtain two inputs which will produce an identical hash value.

Keccak is a cryptographic hash function and the basis for the SHA-3 (Secure Hash Algorithm 3) standard, which was selected by the National Institute of Standards and Technology (NIST) in 2012. Unlike its predecessors, SHA-1 and SHA-2, Keccak employs a unique structure and process, enhancing its security and efficiency [7] [10].

Keccak operates through a series of absorbing and squeezing phases, as indicated in Fig. 1 from [7], where the input data is absorbed into a state array and then permuted. This process involves transforming blocks of 1152, 1088, 832, or 576 bits to generate hash values of 224, 256, 384, and 512 bits. These permutations ensure that even the slightest change in the input data results in a significantly different hash output, making Keccak highly resistant to collisions and attacks [10].

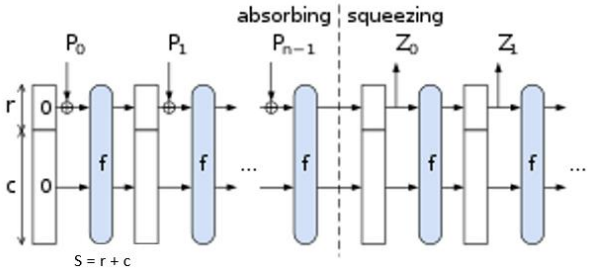


Fig. 1. Absorbing and squeezing phases in Keccak from [7].

C. Chaotic Scheme

Chaos in many contexts often refers to phenomena in which the systems are highly sensitive to initial conditions, resulting in difficult-to-predict behaviour or seemingly random behaviour [11].

In cryptography, a map refers to mathematical functions which link each element from a set to another corresponding element in another set. Maps are utilized to convert data from one form to another.

Chaotic maps are maps which exhibit chaotic properties or uncertainties by producing hard-to-predict numbers. Chaotic maps serve to make cryptographic systems more resistant to attacks due to the difficulty in describing or predicting the chaotic behavioral pattern [5].

D. Logistic-Chebyshev Map

The Logistic-Chebyshev map is a combination of two one-dimensional chaotic maps: the logistic map and the Chebyshev map. By merging these two maps, the Logistic-Chebyshev map is expected to produce more complex and varied results compared to a single chaotic map [3]. This enhanced complexity can be beneficial in various applications, such as in the development of security systems or data scrambling, where chaotic properties are leveraged to generate random number sequences with the desired level of complexity. The following is the Logistic-Chebyshev equation from [3]:

$$X_{n+1} = \left(\alpha X_n(1 - X_n) + \frac{(4-\alpha)}{4} cos(b \times cos^{-1}(X_n))\right) mod\ 1 \quad (1)$$

$$X_n \in (0, 1)$$

$$\alpha \in (0, 4]$$

$$b \in \mathbb{N}$$

E. Sine-Chebyshev Map

The Sine-Chebyshev map is a one-dimensional chaotic system that combines the sine map and the Chebyshev map. The sine map, with its simple mathematical structure, and the Chebyshev map, a polynomial map, both exhibit chaotic behavior and high sensitivity to initial conditions. By integrating these two maps, the Sine-Chebyshev map enhances the unpredictability and complexity of the generated sequences, making it particularly useful for applications in cryptography, random number generation, and other fields requiring high levels of complexity and sensitivity. This synergy creates a more robust and versatile chaotic system. The following is the Sine-Chebyshev equation from [16]:

$$X_{n+1} = \left(\alpha \sin(\pi X_n) + \frac{(4-\alpha)}{4}\cos(b \times \cos^{-1}(X_n))\right) mod\ 1 \quad (2)$$

$$X_n \in (0,1)$$

$$\alpha \in (0,4]$$

$$b \in \mathbb{N}$$

F. DNA Encoding

Deoxyribonucleic Acid (DNA) carries the genetic information of all living organisms, characterized by its four constituent bases: A (Adenine), C (Cytosine), G (Guanine), and T (Thymine). These bases pair uniquely within DNA sequences, where A always pairs with T and C always pairs with G. These pairings form the fundamental structure of DNA, essential for encoding genetic instructions in living cells. In computational applications, these bases can be represented in binary form according to specific encoding rules, facilitating the transformation of plaintext data into a binary format known as DNA encoding. This process is crucial in fields such as cryptography and data security, where the inherent properties of DNA bases can be leveraged to encode information securely and efficiently [13].

DNA decoding, conversely, involves reversing the encoding process to retrieve the original plaintext from its binary representation using the DNA base pairs. This decoding mechanism ensures that the encoded information can be accurately reconstructed into its original form, preserving data integrity and facilitating its practical application in various scientific and technological domains. Overall, DNA encoding and decoding techniques offer a promising approach for secure data transmission and storage, harnessing the natural properties of DNA for innovative computational solutions in information security. Table I indicates sample DNA encoding for two-bit pairs.

TABLE I DNA ENCODING

Rule / Code	1	2	3	4	5	6	7	8
A	00	00	11	11	10	01	10	01
T	11	11	00	00	01	10	01	10
C	10	01	10	01	00	00	11	11
G	01	10	01	10	11	11	00	00

G. S-box

S-box, or Substitution Box, is a component in symmetric cryptography which substitutes values from input data blocks with corresponding outputs based on a certain rule or substitution table. The main purpose of the S-box is to increase the algorithm's security level by complicating statistical analysis and cryptoanalysis attacks.

This research uses S-boxes from the research [4], namely S-box 3, S-box 4, S-box 5, dan S-box 6. S-box 3 is given in Table II below as an example.

TABLE II S-BOX 3

105	197	63	16	136	75	70	74	220	96	100	125	167	98	108	148
242	5	254	93	13	78	253	45	144	12	35	196	226	179	230	44
123	204	15	41	176	0	165	64	11	217	163	59	56	62	134	140
235	250	49	77	131	252	239	157	244	214	129	248	177	113	10	152
103	231	51	130	139	32	73	7	219	33	200	156	146	192	232	191
233	202	187	23	241	246	216	158	31	161	17	94	53	9	206	117
249	89	127	24	195	46	43	162	80	6	48	209	54	119	149	65
92	102	212	135	36	203	28	126	27	132	210	172	85	145	22	224
34	188	50	25	67	225	88	182	84	81	69	240	228	104	143	72
8	86	60	3	171	205	238	55	61	245	79	83	222	58	147	142
121	37	124	4	87	183	1	199	243	166	180	118	114	91	29	184
169	189	110	101	47	170	251	40	186	18	97	229	155	174	236	153
247	150	106	237	168	193	66	2	112	215	14	120	201	21	213	38
95	52	198	57	109	208	178	255	218	211	30	227	190	76	90	82
173	99	42	39	185	194	159	111	138	137	116	181	141	154	160	175
115	68	26	122	20	221	164	71	107	207	234	128	133	151	223	19

H. Mean Squared Error (MSE)

Mean Squared Error is a parameter used to compare the level of pixel difference between two images. A high MSE value indicates that the level of pixel difference between the images is relatively large, meaning the difference between the two images is quite significant. MSE may be obtained from the following equation from [9]:

$$MSE = \frac{1}{XY}\sum_{m=1}^{m=X}\sum_{n=1}^{n=Y}(O(m,n) - R(m,n))^2 \quad (3)$$

I. Peak Signal-to-Noise Ration (PSNR)

Peak Signal-to-Noise Ratio (PSNR) is a parameter used to compare two images. The PSNR value ranges from [0, ∞). The higher the PSNR value, the more similar the compared images are. Conversely, the lower the PSNR value, the more different the compared images are [9]. PSNR is calculated using the following equation from [9]:

$$PSNR = 10 \times \log_{10}\frac{(2^n-1)^2}{MSE} \quad (4)$$

n = bit in the image

MSE = Mean Squared Error value

IV. Analysis and Design

A. Problem Analysis

The primary issue to be addressed in this research is the development of a system which may secure images by means of symmetric cryptography leveraging the Keccak hash function, chaos scheme and DNA encoding.

Functional requirements analysis:

1) System will secure image using the methods mentioned
2) System will secure image with the maximum dimension of 512x512 pixels
3) System will accept input from local file and database
4) System will accept input of secret key to encrypt and decrypt the image
5) System will perform registration and authentication of users
6) System will be able to send encrypted images to the receiver

Non-functional requirements analysis:

1) System will provide a user-interface which is easy to understand for novice users
2) System will validate every user input during encrytion and decryption and provide error message for invalid input.

B. Flowchart

The flowcharts for the activities in this research are described in the following figures.

Fig. 2 describes the proposed algorithm implementation, indicating the important stages in the encryption process starting with the plain image and the secret key: key generation, image decomposition and block permutation, substitution, DNA encoding, complementing, DNA decoding, bitwise operation.

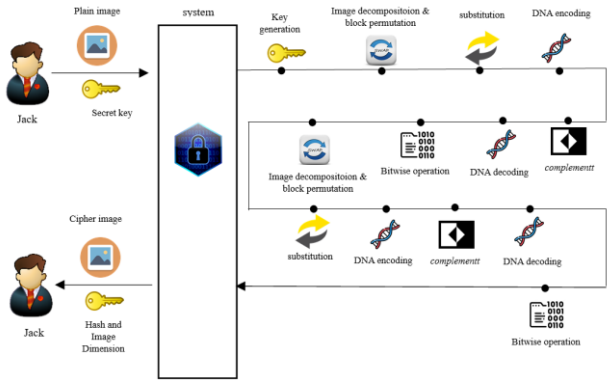


Fig. 2. Encryption process.

Fig. 3 describes the proposed algorithm for the decryption process, which essentially has the same stages as the encryption process but only in reverse order.

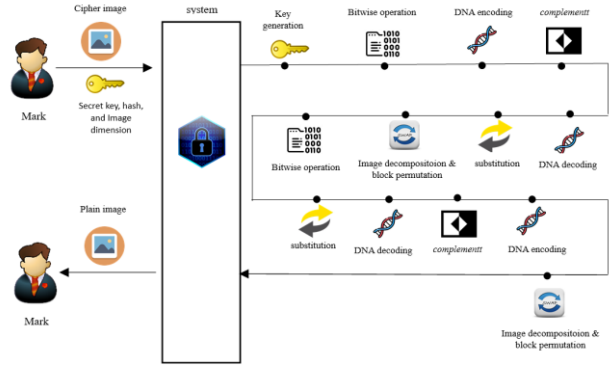


Fig. 3. Decryption process.

C. Proposed Algorithm

1) Key generation

a) Calculate the hash value of plain image *h*.

b) Divide *h* into 32 bytes.

c) Calculate the following, from [4]:

$$X'_0 = \frac{1}{3}\left(X_0 + 2 \times \frac{1}{256} \times bin2dec(h_9 \oplus \ldots \oplus h_{16})\right) \quad (5)$$

$$\alpha'_0 = \frac{1}{3}\left(\alpha_0 + 2 \times \frac{1}{64} \times bin2dec(h_1 \oplus \ldots \oplus h_8)\right) \quad (6)$$

$$X'_1 = \frac{1}{3}\left(X_1 + 2 \times \frac{1}{256} \times bin2dec(h_1 \oplus \ldots \oplus h_{32})\right) \quad (7)$$

$$\alpha'_1 = \frac{1}{3}\left(\alpha_1 + 2 \times \frac{1}{64}\left(\sum_{i=1}^{32} h_1(i)\ mod\ 256\right)\right) \quad (8)$$

d) Calculate Logistic-Chebyshev Map by iteration > 500 times to hide the transient effect. Then, calculate it Mx4N times and put it in matrix. M and N is the image dimension.

e) Map the matrix with 1 if the value > 0.5 and 0 if the value ≤ 0.5.

f) Calculate Sine-Chebyshev Map by iteration > 500 times to hide the transient effect. Then, calculate it MxN times and put it in matrix.

g) Replace each matrix element by calculating

$$Z_1(i,j) = (Y \times 10^{15}) \; mod \; 256 \quad (9)$$

2) Image processing and decomposition

a) Divide image into 64 blocks.

b) Permute these blocks by the following calculation using prime p and h = 1..64

$$b_i = (p \times h) \; mod \; 64 \quad (10)$$

c) Reconstruct image based on the calculation.

3) Substitution

a) Get S-box-3, S-box-4, S-box-5, S-box-6 from [4]

b) Divide the image into 4 parts, substitute each part with each S-box, respectively.

4) DNA encoding

a) Select a DNA rule from the DNA RULE in Table III below.

b) Encode the image using the selected DNA rule, and the image size will be Mx4N.

5) Complement

Using the Logistic Chebyshev matrix, replace the image pixel with the complement if the logistic Chebyshev matrix in the corresponding index is 1.

6) DNA decoding

a) Select a DNA rule from the DNA RULE below.

b) Decode the image using the DNA RULE, restoring the image size back to MxN.

7) Bitwise operation

Using the Sine Chebyshev matrix, XOR the image with the matrix.

8) Calculate MSE and PSNR

a) The MSE parameter is used to determine if the image was successfully encrypted.

b) The PSNR parameter is used to determine if the image was successfully decrypted.

TABLE III DNA RULE

Rule								
Base	1	2	3	4	5	6	7	8
A	00	00	01	01	10	10	11	11
T	11	11	10	10	01	01	00	00
C	01	10	00	11	00	11	01	10
G	10	01	11	00	11	00	10	01

V. IMPLEMENTATION AND TESTING

A. Implementation

In this research, the application was built as a website using Python and Laravel PHP Framework. User data and image data were stored in a MySQL database. The software used for development is Visual Studio Code. This application has five pages: Login, Register, Home, Encryption, Decryption. These pages are indicated in Fig. 4 to Fig. 8. Encrypted images may be stored into the database for access by another user, as indicated in Fig. 7.

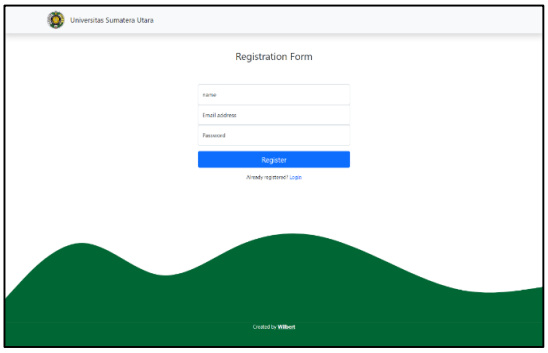


Fig. 4. Registration page.

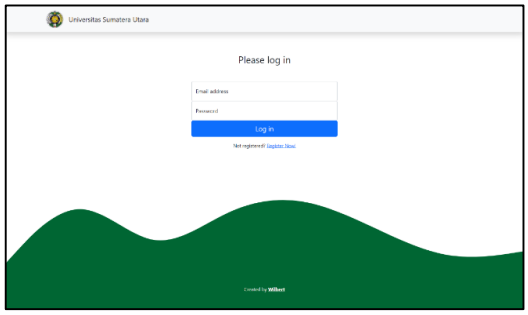


Fig. 5. Login page.

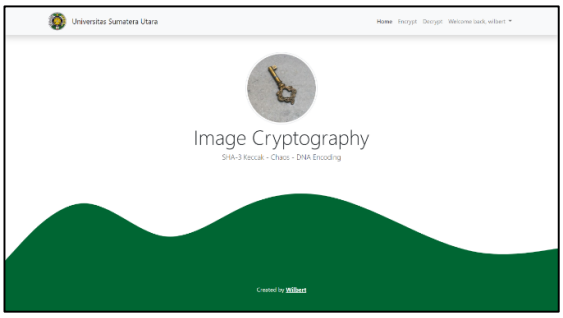


Fig. 6. Home page.

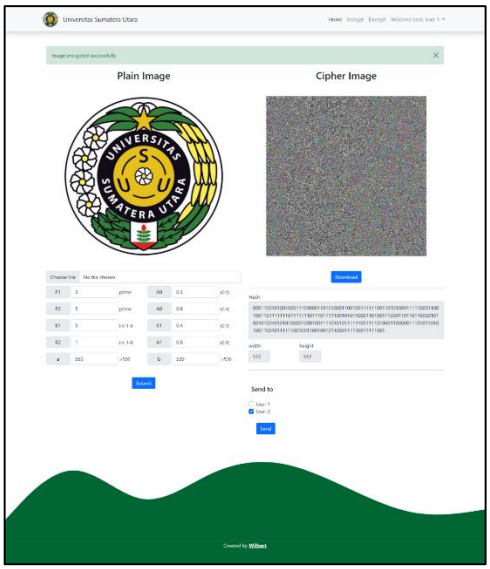


Fig. 7. Encryption page with a sample image.

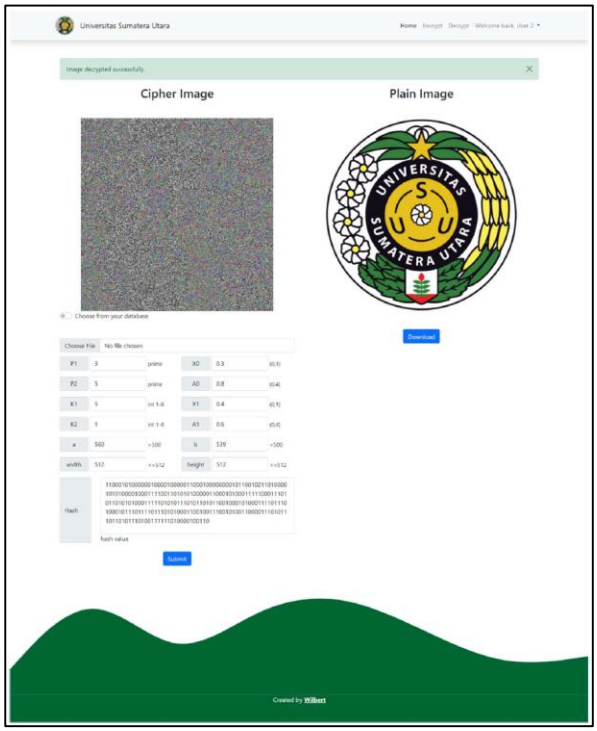


Fig. 8. Decryption page with a sample image.

B. System Testing

Testing consisted of the following steps:

1) Encrypt 5 test images of various sizes with .png extension and different secret keys.

2) Measure the MSE between the original/plain image and the obtained scrambled/cipher image.

3) Decrypt the cipher image back to plain image.

4) Measure the PSNR between the restored image to the original image

Testing was carried out on a laptop with the following specification:

- AMD Ryzen 5 5600H CPU
- GeForce RTX 3050 4GB graphics card
- 16 GB RAM

Required for encryption are the secret keys, which include two prime numbers for image decomposition and block permutation, two random integers between 1 to 8 for DNA encoding and decoding, two random fractions and two random real numbers between 0 and 4 for the chaotic maps. The number of iteration (> 500) for the Logistic-Chebyshev and Sine-Chebyshev mappings are also required. All these values are input manually into the system. The hash value of the selected plain image is calculated initially by the system as well as the dimensions of the image. Parameter for the first test image are indicated in Table IV.

TABLE IV SECRET KEYS FOR FIRST TEST IMAGE

Secret keys	Value	Remarks	Secret keys	Value	Remarks
P1	17	prime	X1	0.7	real (0, 1)
P2	23	prime	A1	3.4	real (0, 4]
K1	8	int 1 – 8	a	765	> 500
K2	2	int 1 – 8	b	594	> 500
X0	0.5	real (0, 1)	Width	400	pixels
A0	2.5	real (0, 4]	Height	400	pixels
Hash	100110010100101110110111011010110101111100111110011010000010000001110110101110111110100110010111000010100011010011010000110110001101011101010000001000000111011010111011111010011001011100001010001101001100010100011010011000101000011011000110101110100000100

Plain image, encrypted image and decrypted image for the second test image are shown in Table VIII. As the case with the first test image, visual inspection showed that the encrypted image is totally different from the plain image, while the decrypted image is virtually the same as the plain image.

TABLE VIII SECOND TEST IMAGE

Plain image	Encrypted Image	Decrypted Image
		

Secret keys used for the encryption and decryption processes are indicated in Table IX. Just as the first test image, secret keys are two prime numbers for image decomposition and block permutation, two integers between 1 and 8 for DNA encoding and decoding, two random fractions and two random real numbers between 0 and 4 for the chaotic maps. The number of iteration (> 500) for the Logistic-Chebyshev and Sine-Chebyshev mappings are also required.

TABLE IX SECRET KEYS FOR SECOND TEST IMAGE

Secret keys	Value	Remarks	Secret keys	Value	Remarks
P1	17	prime	X1	0.12	real (0, 1)
P2	23	prime	A1	2.24	real (0, 4]
K1	2	int 1 – 8	a	599	> 500
K2	8	int 1 – 8	b	652	> 500
X0	0.55	real (0, 1)	Width	383	pixels
A0	3.572	real (0, 4]	Height	512	pixels
Hash	10101100110111100011001011100010011011001001 01101010001011000001011110011111110111011010 00001100110000100010011001110000101101110110 01010010011010111110010111010110100111111110 11101111011100101011101101011011000100110100 10010001101001101101010011010000111				

MSE and PSNR values of the second test image are indicated in Table X. Here the high MSE value of 150.35 and low PSNR value of 27.90 dB between the plain image and encrypted image showed high difference between the two, while low MSE of 0 and high PSNR value of infinity between the decrypted image and plain image, indicated that both images are similar, which again supported the visual inspection result in Table VII.

Processing times for the second test image are indicated in Table XI. Here the processing times are slightly below 15 seconds (14.88 seconds for encryption and 14.97 for decryption).

Table XII shows the result of the third test image. As the case with the second test image, visual inspection showed that the encrypted image is totally different from the plain image, while the decrypted image is virtually the same as the plain image.

Secret keys used for the encryption and decryption of third test image are indicated in Table XIII. Just as the second test image, secret keys consisted of two prime numbers for image decomposition and block permutation, two integers between 1 and 8 for DNA encoding and decoding, two random fractions and two random real numbers between 0 and 4 for the chaotic maps. The number of iteration (> 500) for the Logistic-Chebyshev and Sine-Chebyshev mappings are also required.

TABLE X MSE AND PSNR OF SECOND TEST IMAGE

	Plain image and cipher image	Plain image and decrypted image
MSE	105.35418192545573	0
PSNR (dB)	27.904285817106654	∞

TABLE XI PROCESSING TIME OF SECOND TEST IMAGE

Encryption (seconds)	Decryption (seconds)
14.887351989746094	14.975344181060791

TABLE XII THIRD TEST IMAGE

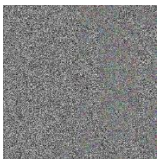
Plain image	Encrypted Image	Decrypted Image
		

TABLE XIII SECRET KEYS FOR THIRD TEST IMAGE

Secret keys	Value	Remarks	Secret keys	Value	Remarks
P1	97	prime	X1	0.758	real (0, 1)
P2	11	prime	A1	1.657	real (0, 4]
K1	5	int 1 – 8	a	1124	> 500
K2	7	int 1 – 8	b	752	> 500
X0	0.5	real (0, 1)	Width	512	pixels
A0	3.21	real (0, 4]	Height	342	pixels
Hash	11101001111000011001110110111101001110011000 10101111111000101000010000010000001000011011 01001000110100101000000000001100010000100011 01000010001011111111111011100101110101110001 00101010101011011101110110110011100100011011 11001001100000011101100101111011111				

MSE and PSNR values of the third test image are indicated in Table XIV. Here the high MSE value of 150.53 and low PSNR value of 27.89 dB between the plain image and encrypted image showed the high difference between the two, while low MSE of 0 and high PSNR value of infinity between decrypted image and plain image, indicated that both images are similar, which again supported the visual inspection result in Table XII.

Processing times for the third test image are indicated in Table XV. Here the encryption processing time was 14.83 seconds, and decryption was 15.19 seconds.

Table XVI shows the result of the fourth test image. As the case with the previous test images, Table XVII contains the secret keys, while Table XVIII shows the MSE and PSNR values between the plain image and cipher image, and the plain image and the decrypted image. Table XIX indicate the processing times for encryption and decryption.

Table XX shows the result of the fifth test image. As the case with the previous test images, Table XXI contains the secret keys, while Table XXII shows the MSE and PSNR values between the plain image and cipher image, and the plain image and the decrypted image. Table XXIII indicate the processing times for encryption and decryption.

TABLE XIV MSE AND PSNR OF THIRD TEST IMAGE

	Plain image and cipher image	Plain image and decrypted image
MSE	105.53628412882487	0
PSNR (dB)	27.89678562013698	∞

TABLE XV PROCESSING TIME OF THIRD TEST IMAGE

Encryption (seconds)	Decryption (seconds)
14.82814908027649	15.187212944030762

TABLE XVI FOURTH TEST IMAGE

Plain image	Encrypted Image	Decrypted Image
		

TABLE XVII SECRET KEYS FOR FOURTH TEST IMAGE

Secret keys	Value	Remarks	Secret keys	Value	Remarks
P1	17	prime	X1	0.01	real (0, 1)
P2	37	prime	A1	2.2	real (0, 4]
K1	2	int 1 – 8	a	582	> 500
K2	1	int 1 – 8	b	725	> 500
X0	0.99	real (0, 1)	Width	512	pixels
A0	4	real (0, 4]	Height	394	pixels
Hash	11011000011101100010001110011111010110111100 00100010011110110110111111011100100010101000 00010010010000111110101110100100111011110110 01100011100111010110000110000000011010010110 00110000010000010110000110011001001001110110 00000011111101000000110111011001 1010				

TABLE XVIII MSE AND PSNR OF FOURTH TEST IMAGE

	Plain image and cipher image	Plain image and decrypted image
MSE	105.44661966959636	0
PSNR (dB)	27.900476988632636	∞

TABLE XIX PROCESSING TIME OF FOURTH TEST IMAGE

Encryption (seconds)	Decryption (seconds)
15.099057912826538	15.182414054870605

TABLE XX FIFTH TEST IMAGE

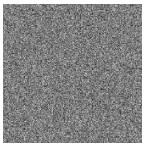
Plain image	Encrypted Image	Decrypted Image
		

TABLE XXI SECRET KEYS FOR FIFTH TEST IMAGE

Secret keys	Value	Remarks	Secret keys	Value	Remarks
P1	11	prime	X1	0.01	real (0, 1)
P2	37	prime	A1	4	real (0, 4]
K1	1	int 1 – 8	a	501	> 500
K2	8	int 1 – 8	b	501	> 500
X0	0.99	real (0, 1)	Width	235	pixels
A0	4	real (0, 4]	Height	305	pixels
Hash	00000010010000001100110100001001111110010010 1 00101000000010101011001101101000110100110 1100 10010010111101100001100110000110101100001011 0 11110100101010011000001001011001010001100000 0 00000010010101111100111011101101101001011010 0 00000100110111111001011110010000				

TABLE XXII MSE AND PSNR OF FIFTH TEST IMAGE

	Plain image and cipher image	Plain image and decrypted image
MSE	105.21969985961914	0
PSNR (dB)	27.90983302234912	∞

TABLE XXIII PROCESSING TIME OF FIFTH TEST IMAGE

Encryption (seconds)	Decryption (seconds)
14.929418087005615	17.71390175819397

VI. DISCUSSION

Several additional observations maybe obtained from the implementation of the cryptosystem, as follows:

- User and image data were stored in the system using a MySQL Database
- All input images sizes were padded to achieve a dimension of 512x512 pixel.
- High MSE values (around 150) and low PSNR values (around 27 dB) between the plain images and the encrypted images indicated the plain images were successfully converted into abstract images.

- MSE values of 0 and high PSNR values (infinity) between the plain images and the decrypted images indicated that the encrypted images were successfully restored to images identical to the original plain images.
- Processing times for encryption and decryption were around 15 seconds on the testing laptop.

VI. Conclusion

Based on the results obtained from the above system testing, it may be concluded that:

1) The algorithms used were capable of performing image encryption and decryption effectively, building on previous studies mentioned at the beginning of this study. This system has successfully integrated the SHA-3 Keccak hash algorithm, Chaos Scheme, and DNA Encoding.

2) The system was implemented into a website using Python, PHP, and a MySQL database.

3) The evaluation of MSE and PSNR values showed that the algorithm successfully abstracted the images and restored them to their original form.

4) The processing time required for each encryption and decryption operation is still quite high for the specified laptop used in system testing.

5) This study has contributed to the field of cryptography an alternate approach to image security using the aforementioned algorithms.

VII. Future Work

Limitations of the present implementation include a symmetric cryptographic approach (which may introduce a key distribution problem), while also limiting images to be encrypted having a maximum dimension of 512x512 pixels. This system is also run on a stand-alone PC-based platform. Secret keys required for the encryption and decryption were input manually, resulting in a not very large key space. Several approaches for future work maybe suggested: firstly, secret keys maybe generated by the system on-demand, therefore ascertaining better randomness and larger key space; secondly, to address key distribution problem faced by symmetric cryptosystem by combining the present algorithms with a public key algorithm such as Extended RSA to achieve a more secure hybrid cryptosystem; thirdly, improving so that images to be encrypted are not limited to a dimension of 512x512 pixels; fourth, to port the cyptosystem to a mobile platform such as Android or iOS, and lastly integrating the present system into a communication system such as an e-mail or instant messaging application.

References

[1] A. Belazi, M. Talha, S. Kharbech and W. Xiang, "Novel Medical Image Encryption Scheme Based on Chaos and DNA Encoding," in IEEE Access, vol. 7, pp. 36667-36681, 2019, doi: 10.1109/ACCESS.2019.2906292.

[2] Alrubaie, A.H., Khodher, M.A.A. & Abdulameer, A.T. “Image encryption based on 2DNA encoding and chaotic 2D logistic map”. *J. Eng. Appl. Sci.* **70**, 60 (2023). https://doi.org/10.1186/s44147-023-00228-2.

[3] B. Abd-El-Atty, M. Amin, A. Abd-El-Latif, H. Ugail and I. Mehmood, "An Efficient Cryptosystem based on the Logistic-Chebyshev Map," 2019 13th International Conference on Software, Knowledge, Information Management and Applications (SKIMA), Island of Ulkulhas, Maldives, 2019, pp. 1-6, doi: 10.1109/SKIMA47702.2019.8982535.

[4] I. Hussain, T. Shah, M. A. Gonda *et al*, “An efficient approach for the construction of LFT S-boxes using chaotic logistic map”, *Nonlinear Dyn* **71**, 133–140 (2013). https://doi.org/10.1007/s11071-012-0646-1.

[5] J. Arif et al., "A Novel Chaotic Permutation-Substitution Image Encryption Scheme Based on Logistic Map and Random Substitution," in IEEE Access, vol. 10, pp. 12966-12982, 2022, doi: 10.1109/ACCESS.2022.3146792.

[6] J. -Philippe Aumasson, Serious Cryptography, San Francisco: No Starch Press, Inc., 2018.

[7] M. A. Patil and P. T. Karule, "Design and implementation of keccak hash function for cryptography," 2015 International Conference on Communications and Signal Processing (ICCSP), Melmaruvathur, India, 2015, pp. 0875-0878, doi: 10.1109/ICCSP.2015.7322620.

[8] M. T. Suryadi, Z. Rustam, W. Widhianto, “Implementasi Algoritma Enkripsi Citra Digital Menggunakan Skema Transposisi Berbasis Fungsi Chaos”, Prosiding Seminar Ilmiah Nasional Komputer dan Sistem Intelijen (KOMMIT), vol. 8, pp. 376-380, Depok, 2014.

[9] N. Mahendiran, C. Deepa, “A Comprehensive Analysis on Image Encryption and Compression Techniques with the Assessment of Performance Evaluation Metrics”, *SN COMPUT. SCI.* **2**, 29 (2021). https://doi.org/10.1007/s42979-020-00397-4.

[10] P. P. Pittalia, “A Comparative Study of Hash Algorithms in Cryptography”. International Journal of Computer Science and Information Technology, Vol. 8, Issue. 6, pp.147-152, 2019.

[11] R. Munir, “Algoritma Enkripsi Citra dengan Pseudo One-Time Pad yang Menggunakan Sistem Chaos”, Konferensi Nasional Informatika (KNIF), pp. 12-16, 2011.

[12] S. Kromodimoeljo, Teori dan Aplikasi Kriptografi, SPK IT Consulting, 2009.

[13] T. A. S. Yusri and D. Rudhistiar, “Enkripsi Citra Digital Berbasis Kombinasi Arnold Cat Map Termodifikasi dan DNA Encoding”, Mnemonic, vol. 5, no. 2, pp. 173-177, Aug. 2022.

[14] W. Stallings, Cryptography and Network Security: Principles and Practice, 7th Edition, Pearson, 2016.

[15] X. Wang, Y.Su, “Image encryption based on compressed sensing and DNA encoding”, in Signal Processing: Image Communication, Volume 95, 2021, 116246, ISSN 0923-5965, https://doi.org/10.1016/j.image.2021.116246.

[16] Y. Zhong, H. Liu, R. Lan, T. Wang, X. Sun and X. Luo, "2D Chebyshev-Sine Map for Image Encryption," 2018 7th International Conference on Digital Home (ICDH), Guilin, China, 2018, pp. 1-8, doi: 10.1109/ICDH.2018.000.

Architecting a Privacy-Focused Bitcoin Framework Through a Hybrid Wallet System Integrating Multiple Privacy Techniques

Lamiaa Said[1], Hatem Mohamed[2], Diaa Salama[3], Nesma Mahmoud[4]
Department of Information System-Faculty of Computers and Artificial Intelligence, Benha University, Egypt[1, 3]
Department of Information System-Faculty of Computers and Information, Menoufia University, Egypt[2, 4]

***Abstract*—Although Bitcoin enables pseudonymous peer-to-peer digital transactions, its transparent public ledger architecture allows for blockchain analysis that can compromise user anonymity. Despite the presence of wallets with privacy-enhancing features, no single solution currently offers comprehensive anonymity independently. Existing privacy-preserving techniques such as CoinJoin, PayJoin, and Stealth Addresses offer differing degrees of anonymity, yet each exhibits intrinsic limitations. This study proposes a hybrid privacy architecture that integrates multiple privacy-enhancing techniques into a unified and coherent transaction workflow. By integrating decentralized CoinJoin mixing, PayJoin for input ownership obfuscation, and Stealth Addresses for unlinkable payments, the proposed model establishes a robust, privacy-oriented framework for Bitcoin transactions. The framework is implemented and evaluated through pre-funded Sparrow and JoinMarket wallets, interconnected via a fully synchronized Bitcoin Core node deployed on the testnet environment. All communications are routed via the Tor network to maintain anonymity at the network layer. Using testnet-based simulations, we evaluate the effectiveness of the architecture. The results show that combining these techniques substantially strengthens resistance to common deanonymization heuristics, enhances transaction unlinkability, and achieves higher overall anonymity than relying on individual methods alone. This demonstrates the synergistic effect of the hybrid model in providing more resilient protection against transaction tracing and blockchain surveillance.**

***Keywords*—Bitcoin; privacy; anonymity; wallet; blockchain; Coinjoin; Payjoin; stealth address**

I. Introduction

Bitcoin was initially conceived as a peer-to-peer electronic cash system, enabling direct transactions between users without the involvement of centralized intermediaries, and theoretically offering a high degree of privacy [1]. However, Bitcoin's inherent transparency has emerged as a significant privacy concern over time Despite employing pseudonymous addresses, all Bitcoin transactions are permanently recorded on a publicly accessible blockchain [2]. Although Bitcoin substitutes real-world identifiers with pseudonymous address formats, this mechanism alone is insufficient to guarantee true anonymity [3]. Over the years, a wide range of blockchain forensic techniques has been developed to exploit this transparency—leveraging transaction structures, address reuse patterns, and input-output correlations to trace user behaviors and cluster wallet activities for deanonymization. As Bitcoin adoption continues to grow, adversarial deanonymization strategies have evolved in parallel, aiming to link blockchain activity with real-world identities [4].

In response to these privacy threats, the Bitcoin ecosystem has introduced a variety of privacy-enhancing mechanisms designed to mitigate the risks associated with transaction transparency. For instance, CoinJoin allows multiple users to aggregate their transaction inputs and outputs into a single transaction, thereby obscuring direct associations between them [5]. This mixing methodology is practically implemented by the JoinMarket wallet. Other wallets, such as Sparrow, incorporate additional privacy features such as PayJoin—an interactive transaction protocol designed to further obscure input ownership heuristics [6]. Additionally, privacy tools like Stealth Addresses and PayNym support reusable payment codes that prevent the disclosure of linkable public addresses [7]. At the network layer, anonymity can be reinforced by routing transactions through the Tor network, while operating a self-hosted Bitcoin Core node prevents exposure of IP addresses during transaction broadcasting [8]. Although each one of these techniques enhances privacy to a certain extent, no one individually provides comprehensive protection. Dependence on any single privacy method frequently leaves users susceptible to forms of blockchain and network-level analysis.

This situation reveals a critical research gap as most prior studies evaluate these privacy techniques in isolation, often highlighting their strengths but overlooking their inherent limitations when deployed alone. There has been limited exploration of how these techniques could be combined into a cohesive, layered framework capable of countering multiple deanonymization strategies simultaneously.

To address these limitations, we propose a hybrid wallet architecture that consolidates multiple privacy-preserving mechanisms across distinct wallets and tools. Our model employs funded Sparrow Wallets to initiate PayJoin transactions and manage stealth address payments, JoinMarket to conduct decentralized CoinJoin mixing, and a fully synchronized Bitcoin Core testnet node as its infrastructural backbone. All system components are interconnected via the Tor network to maintain robust anonymity at the network communication layer. By integrating these tools, we construct a layered privacy architecture designed to confound multiple levels of forensic and heuristic analysis [9].

Accordingly, this study focuses on evaluating whether the integration of multiple privacy-enhancing techniques within a unified hybrid wallet model can provide stronger anonymity guarantees than the isolated use of individual methods.

This study investigates whether a hybrid approach can provide stronger anonymity guarantees compared to the isolated application of individual privacy techniques. Through the implementation and testing of this framework using real-world wallets and a Bitcoin Core testnet node, we demonstrate that integrating existing privacy tools significantly enhances the resilience and anonymity of Bitcoin transactions. Our results indicate that integrating multiple privacy techniques can substantially improve user anonymity within the Bitcoin ecosystem.

The rest of the paper is organized as follows: Section II provides background and related work, detailing the foundational necessary concepts and explores existing privacy tools and previous research. Section III outlines the proposed framework, describing the experimental setup, tools, and tests used in the study. Section IV presents experimental results and key observations. Finally, Section V the conclusion and future work, summarizing the findings and discussing Promising directions for future work.

II. Background and Related Work

In 2009, Bitcoin was introduced by Satoshi Nakamoto as a decentralized alternative to traditional fiat systems, enabling peer-to-peer value transfers without reliance on financial intermediaries. Each Bitcoin transaction is immutably recorded on a publicly accessible ledger known as the blockchain, allowing full transparency to all participants [1].

A. Bitcoin's Pseudonymity

Bitcoin enables user interaction via alphanumeric addresses instead of verifiable legal identities, thereby offering pseudonymity rather than full anonymity. Despite the absence of explicit personal identifiers, advanced analytic techniques can correlate addresses, cluster user activity, and trace fund flows across the blockchain. The pseudonymity provided by Bitcoin is frequently inadequate for users demanding higher levels of privacy [10]. As a response, numerous privacy-preserving techniques have been developed and deployed, each offering distinct advantages and facing unique limitations. Among the most implemented methods are CoinJoin, PayJoin, and Stealth Addresses. Thus, although Bitcoin does not explicitly disclose user identities, it lacks robust anonymity, particularly when subjected to sophisticated forensic analysis techniques [11].

B. CoinJoin and Decentralized Mixing

CoinJoin, introduced by Greg Maxwell in 2013 [12], is a Bitcoin privacy-enhancing technique designed to obscure transaction origins. The technique aggregates inputs from multiple users into a single transaction, then redistributes them as outputs typically of equal or standardized value. This obfuscation makes it significantly more difficult for observers to infer input-output mappings. Empirical studies validate CoinJoin's effectiveness in enlarging the anonymity set; however, vulnerabilities remain, timing analysis and behavioral patterns among participants [5].

C. PayJoin (P2EP) Input Ownership Confusion

PayJoin, or Pay-to-EndPoint, enhances privacy by making the payment transaction appear as if it involves two unrelated parties contributing inputs. This approach disrupts conventional input ownership heuristics, which typically assume all transaction inputs originate from the sender. Successful execution of PayJoin necessitates that both sender and receiver be simultaneously online and utilize wallets that support the protocol. Despite its advantages, PayJoin adoption remains limited due to synchronization challenges and compatibility constraints across different wallet implementations [6].

D. Stealth Addresses (PayNym) and Unlinkability

Stealth Addresses represent another vital privacy mechanism, designed to prevent address reuse and facilitate unlinkable transactions. This method enables the sender to generate a unique, one-time destination address for each transaction, derived from a public stealth address provided by the recipient. Consequently, each payment appears to target a distinct destination, effectively nullifying address reuse heuristics [13]. PayNym extends the stealth address paradigm by introducing reusable payment codes, allowing multiple unlinkable payments to be conducted with enhanced convenience. Only the sender and intended recipient can deterministically compute the correct destination address, thereby enhancing privacy without compromising convenience. Although stealth address support in Bitcoin remains relatively uncommon, its adoption has been revitalized in wallets such as Sparrow [7].

E. Network-Layer Privacy with Tor

Network-layer privacy using Tor adds a crucial layer of protection to Bitcoin transactions by concealing the origin and destination of network messages. maintains online anonymity by routing traffic through multiple layers of encrypted relays. Each relay hop applies an additional layer of encryption, making it difficult for observers to trace the source or destination of the traffic. Since each Tor node is aware only of its adjacent nodes, no single observer can reliably correlate a user's IP address with a specific Bitcoin transaction or node [8].

F. Bitcoin Full Node

A Bitcoin full node serves as a fundamental network component, maintaining a comprehensive and continuously updated copy of the entire blockchain [14]. Unlike lightweight client wallets, full nodes autonomously validate all transactions and blocks in accordance with Bitcoin's consensus protocol. This architecture grants users direct, trustless access to the blockchain, eliminating the need for third-party intermediaries. Operating a full node enhances user privacy and simultaneously contributes to the network's decentralization and security. For research and experimentation, testnet full nodes provide a secure sandbox environment to study transaction dynamics, wallet functionality, and privacy features without risking actual Bitcoin assets [15].

G. Bitcoin Wallets

A Bitcoin wallet is either a software or hardware solution that facilitates the secure storage, transmission, and receipt of Bitcoin. Rather than storing physical coins, wallets manage private keys—cryptographic credentials that grant control over

Bitcoin addresses and associated funds [16]. Some wallets integrate with full Bitcoin nodes for autonomous transaction verification, whereas others depend on third-party servers. Beyond core functionalities, many contemporary wallets incorporate privacy-enhancing features such as CoinJoin, PayJoin, and Stealth Addresses to mitigate blockchain surveillance and user traceability. The choice of wallet significantly influences a user's privacy posture, security robustness, and degree of control over Bitcoin assets [17].

- JoinMarket Wallet JoinMarket is a privacy-centric Bitcoin wallet that provides a decentralized implementation of the CoinJoin mixing protocol. It enables users to mix their coins with others, thereby obscuring transactional origins and enhancing blockchain-level privacy [18].
- Sparrow Wallet is a feature-rich desktop Bitcoin application that combines an intuitive interface with advanced functionalities such as PayJoin and Stealth Address support. It's designed for users who want more control over their transactions and better privacy options, designed for both beginners and experienced users. It is tailored for users seeking granular control over transactions alongside enhanced privacy capabilities [19].

H. Hybrid and Layered Privacy Model

While each individual technique provides only partial protection, recent research in blockchain privacy emphasizes the effectiveness of layered models in achieving stronger anonymity guarantees. For instance, [20] proposed a scheme that combines stealth addresses with Zcash-style note commitments to effectively conceal recipient identities and transaction amounts. An empirical study in [5] demonstrated that, despite mixing, decentralized CoinJoin wallets remain vulnerable to on-chain heuristics and suggested that augmenting CoinJoin with Lightning Network privacy measures can significantly enhance anonymity. Additionally, [21] demonstrated that integrating CoinJoin with network-layer obfuscation techniques such as Tor or Dandelion++ can substantially strengthen resistance to surveillance efforts. [22] emphasized the persistent struggle between privacy-enhancing technologies and blockchain forensic techniques. Their findings underscore the necessity of continuously evolving flexible and adaptive privacy strategies to preserve transaction anonymity in adversarial environments. More recently, several studies further highlight advancements and remaining gaps in layered privacy frameworks. The Springer survey [23] explored cryptography-based approaches to blockchain privacy, highlighting potential anonymity improvements but limited network-layer protections. In study [24] proposed a homomorphic-encryption-enhanced stealth address protocol (HE-DKSAP), offering strong address-level privacy while still lacking integrated network anonymity. The study [25] provided a comprehensive survey of cryptocurrency mixing techniques, emphasizing the benefits of combining multiple approaches but noting challenges in usability, adoption, and practical implementation. These findings collectively underscore the need for operational hybrid frameworks that integrate multiple privacy mechanisms across both blockchain and network layers.

I. Limitations of Previous Privacy-Preserving Approaches and Research Gap

Although significant progress has been made in developing privacy-preserving techniques for Bitcoin, prior research largely evaluates these methods in isolation. As a result, each approach remains exposed to targeted forensic strategies that exploit its specific weaknesses. For instance, CoinJoin effectively enlarges the anonymity set but can still be undermined by timing or clustering analysis. Similarly, PayJoin disrupts conventional ownership heuristics yet suffers from adoption and synchronization challenges, while stealth addresses prevent address reuse but face limited wallet integration and usability concerns. Network-layer protections such as Tor provide communication anonymity but remain vulnerable to advanced traffic correlation.

To clarify these observations, Table I summarizes the strengths and limitations of key privacy approaches when applied individually.

TABLE I. STRENGTHS AND LIMITATIONS OF INDIVIDUAL PRIVACY-PRESERVING APPROACHES IN BITCOIN

Technique / Area	Strengths	Limitations (when used alone)
CoinJoin (Decentralized Mixing)	Enlarges anonymity set; obfuscates input-output links	Vulnerable to timing and heuristic analysis
PayJoin (P2EP)	Breaks ownership heuristics; improves input confusion	Requires sender-receiver synchronization; limited adoption
Stealth Addresses / PayNym	Prevents address reuse; enables unlinkable payments	Sparse adoption; usability challenges
Network-layer privacy (Tor)	Masks IP addresses; prevents direct network tracing	Still exposed to advanced traffic correlation
Full Nodes	Trustless validation; independence from intermediaries	Resource-intensive; not directly enhancing anonymity

These limitations indicate that no single technique can independently deliver comprehensive anonymity. Existing studies highlight the strengths of each method but stop short of exploring a consolidated approach that leverages their complementary benefits while mitigating weaknesses. To address this gap, the present work proposes a hybrid wallet architecture that integrates CoinJoin, PayJoin, Stealth Addresses, and Tor into a unified framework. By combining these methods atop a Bitcoin Core full node, the system establishes layered defenses against blockchain, transaction, and network-level deanonymization, thereby advancing beyond prior single-technique approaches.

III. PROPOSED FRAMEWORK

This section outlines the implementation of the proposed hybrid privacy model, which leverages a combination of real-world Bitcoin wallet applications, full node infrastructure, and anonymity-preserving network technologies. All components were deployed within a Bitcoin testnet environment to facilitate realistic, secure, and risk-free experimentation. The primary objective was to integrate multiple privacy-enhancing techniques into a unified transaction workflow that optimizes anonymity without compromising usability or system

functionality. The implementation incorporates four core privacy techniques: CoinJoin, PayJoin, Stealth Addresses, and Tor-based network obfuscation.

A. System Overview

The hybrid privacy architecture was structured as a layered system composed of four principal components shown in Fig. 1.

1) Wallet layer (application layer privacy): This layer handles transaction creation, signing, and coordination using JoinMarket and Sparrow Wallet. Enable users to manage privacy settings such as CoinJoin, PayJoin, Stealth Addresses, and coin control, giving them flexible control over their transaction-level privacy.

2) Transaction layer (on-chain privacy): To obfuscate sender-receiver relationships, break common blockchain heuristics, and prevent linkability across transactions using these techniques:

- CoinJoin for collaborative transaction mixing.
- PayJoin to obscure input ownership.
- Stealth Addresses to prevent address reuse.

3) Network layer (network-level anonymity): A locally configured Tor service hides IP addresses and location metadata when wallets or nodes communicate over the Bitcoin network to Prevent network observers or blockchain surveillance firms from linking transactions to users' identities or locations.

4) Infrastructure layer (node-level control and testing): A fully synchronized Bitcoin Core testnet, providing blockchain access and transaction validation and broadcasting transactions privately without relying on third-party servers or light clients.

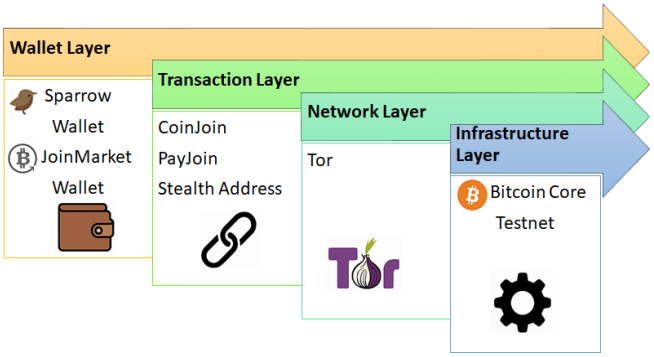


Fig. 1. The layers of the hybrid privacy framework.

This architecture facilitated testing of privacy-preserving transactions across interconnected wallet systems and layers.

B. Software and Tools

The implementation and evaluation of the proposed privacy architecture employed the following open-source software tools, as detailed in Table II.

All software components were deployed on a Windows 10 platform powered by an Intel Core i7 processor, 8 GB of RAM, and a stable broadband internet connection. Tor and Bitcoin Core were executed as background services, while wallet applications were locally configured to communicate via Remote Procedure Call (RPC) interfaces and proxy routing.

TABLE II. SOFTWARE USED IN THE HYBRID PRIVACY FRAMEWORK

Component	Version	Functionality
Bitcoin Core	v25.0 (testnet)	Full-node implementation for transaction validation and verified blockchain synchronization in the testnet environment
JoinMarket	v0.9.11	Decentralized wallet implementing the CoinJoin protocol to perform collaborative transaction mixing
Sparrow Wallet	v1.7.9	Graphical wallet supporting PayJoin, PayNym (Stealth Addresses), and testnet operations, designed for privacy-focused transaction management
Tor Daemon	v0.4.8	Onion-routing service providing network-layer anonymity by concealing IP addresses and communication paths

Bitcoin Testnet functions as a parallel blockchain designed explicitly for development, testing, and experimentation without incurring financial risk. Because Testnet coins lack real-world monetary value, they allow unrestricted experimentation and system evaluation without financial consequences.

C. Data Analysis Tools

- Blockchain Explorers (e.g., Blockstream Explorer [26], Mempool Testnet [27]): Utilized to monitor transaction flows and conduct on-chain activity analysis within the Bitcoin testnet environment.
- Network Sniffers (e.g., Wireshark [28]): Employed to capture and inspect network-level packet data, validating that all transaction communications were successfully routed through the Tor network for anonymity verification.

D. Bitcoin Core Testnet Configuration

A dedicated Bitcoin Core node was deployed on the testnet network to facilitate secure and isolated transaction testing. The configuration settings below were specified in the bitcoin.conf file to support testnet functionality:

bitcoin.conf	
testnet=1	Activates the testnet mode for safe experimentation
server=1	Enables the node to accept RPC commands
rpcuser=bitcoinrpc rpcpassword=***	Specifies RPC authentication credentials
rpcport=18332 rpcbind=127.0.0.1	Binds RPC access to the local machine via port 18332
txindex=1	Enables a full transaction index to support advanced queries
proxy=127.0.0.1:9050	Routes all node communications through the local Tor proxy
onlynet=onion	Restricts network connections to Tor-only peers for maximum anonymity

This configuration enabled Sparrow Wallet and JoinMarket to interact directly with the blockchain—querying data, constructing transactions, and broadcasting them securely over the testnet network through the local Tor proxy.

E. JoinMarket Wallet Setup

JoinMarket was configured to operate in testnet mode and used to perform CoinJoin mixing transactions.

The setup process included the following key steps:

- Creating new JoinMarket wallets tailored for testnet experimentation.
- Executing CoinJoin mixing sessions with testnet coins to obscure transactional origins.
- Enabling Tor integration in the joinmarket.cfg file to route communications anonymously via the Tor network.

joinmarket.cfg	
network = testnet	Specifies the use of Bitcoin testnet for safe testing.
blockchain_source = bitcoin-rpc	Connects JoinMarket to Bitcoin Core via RPC for blockchain data access.
rpc_user = bitcoinrpcrpc_ password = 12345 rpc_host = 127.0.0.1 rpc_port = 18332	Defines RPC credentials and local host settings to securely interface with the Bitcoin Core node.
rpc_wallet_file = jmwallet	Points to the specific bitcoin core wallet file connected to JoinMarket.
use_tor = true socks5_host = 127.0.0.1 socks5_port = 9050	Enables Tor routing for privacy-preserving transaction broadcasting.

CoinJoin transactions were conducted with a configurable number of participants, thereby increasing the size of the anonymity set and enhancing transaction obfuscation.

F. Sparrow Wallet Configuration

Sparrow Wallet served as the primary user interface for initiating PayJoin transactions and handling Stealth Address-based payments. The wallet was integrated with the local Bitcoin Core testnet node, as depicted in Fig. 2, to enable secure and private transaction execution.

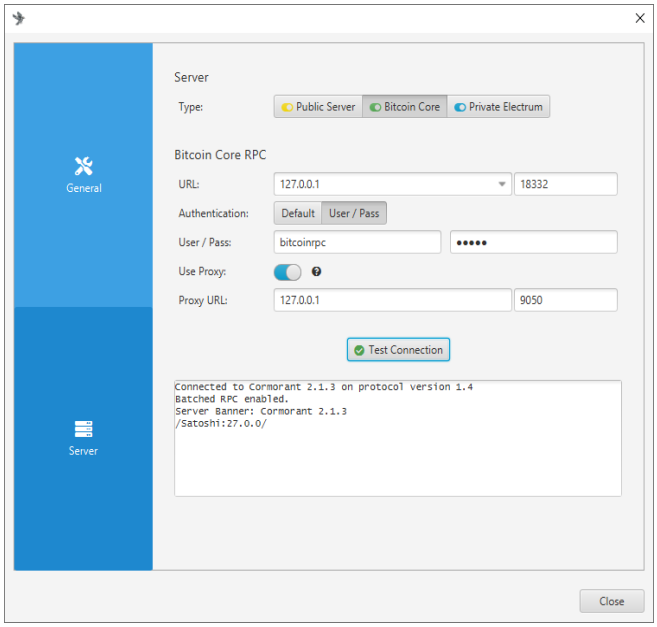


Fig. 2. Sparrow wallet configuration.

Privacy Features Enabled:

1) PayJoin (P2EP): Enabled within Sparrow to facilitate interactive transactions in which both sender and recipient provide inputs confounding traditional ownership heuristics.

2) Stealth Addresses (PayNym): Configured to generate unique, unlinkable payment addresses using reusable public identifiers, preserving receiver privacy.

3) Tor Integration: Enabled to route all network traffic through a local Tor proxy (127.0.0.1:9050), ensuring full communication anonymity.

G. Tor Network Routing

The Tor network was employed to mitigate metadata leakage and preserve user anonymity at the network communication layer. The following parameters were configured in the torrc file to enable secure onion routing for all wallet and node communications:

Torrc	
SocksPort 9050	Designates the SOCKS proxy port used by applications (e.g., wallets) to tunnel traffic through Tor.
ControlPort 9051	Opens a control interface allowing software to manage or monitor the Tor Daemon.
CookieAuthentication 1	Enables secure authentication using cookie-based access control to the Tor ControlPort.

The successful operation of Tor was validated by inspecting system logs and confirming that all wallet interactions were routed through onion services or hidden service addresses.

H. Workflow Integration

The integrated privacy mechanisms were orchestrated through a multi-stage transaction workflow, as illustrated in Fig. 3.

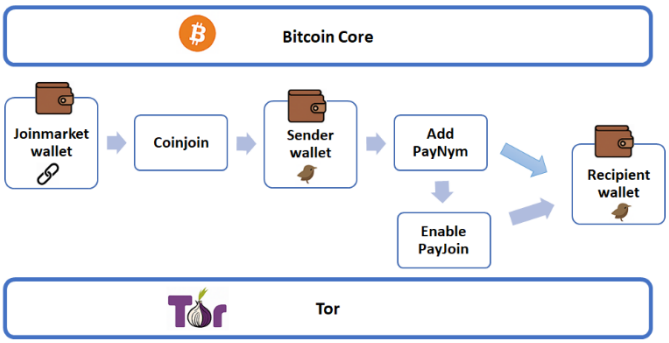


Fig. 3. Multi-stage transaction flow

The multi-stage transaction was implemented in the following sequence:

Step 1: Launch Bitcoin Core in Testnet mode (Fig. 4), ensuring the node is fully synchronized and all wallets are properly connected.

Step 2: Sparrow Wallet was Installed and configured with two pre-funded testnet wallets representing sender and recipient roles.

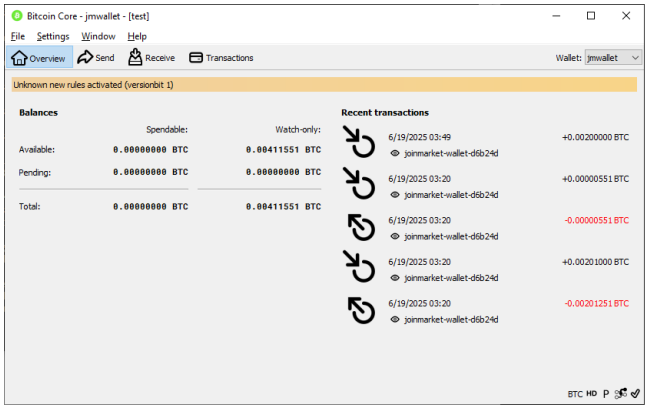


Fig. 4. Bitcoin Core in testnet mode.

Step 3: CoinJoin mixing via JoinMarket for eliminating direct traceability to the coin source and generating clean, unlinkable UTXOs.

As depicted in Fig. 5, a new JoinMarket wallet was initialized, and testnet coins were obtained via a faucet [29].

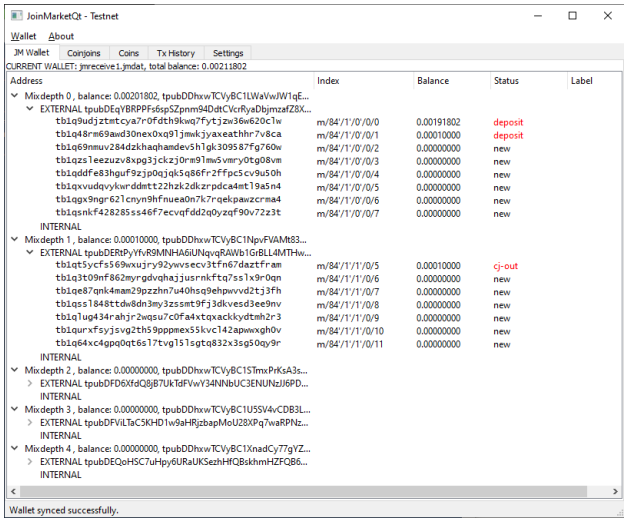


Fig. 5. JoinMarket wallet.

A CoinJoin transaction was initiated with a specified target amount and selected counterparties, as shown in Fig. 6.

The resulting anonymized CoinJoin output UTXOs were transferred to a new address in the Sparrow Sender Wallet to serve as clean inputs for the next step.

Step 4: PayNym Initialization to prevent address reuse and enhance recipient identity privacy.

The recipient published their PayNym identifier, illustrated in Fig. 7.

The sender imported the PayNym into their Sparrow Wallet (Fig. 8) to establish a secure communication link.

An initial broadcast transaction was used to establish a shared secret, after which the sender could derive one-time stealth addresses linked to the recipient's PayNym for each payment.

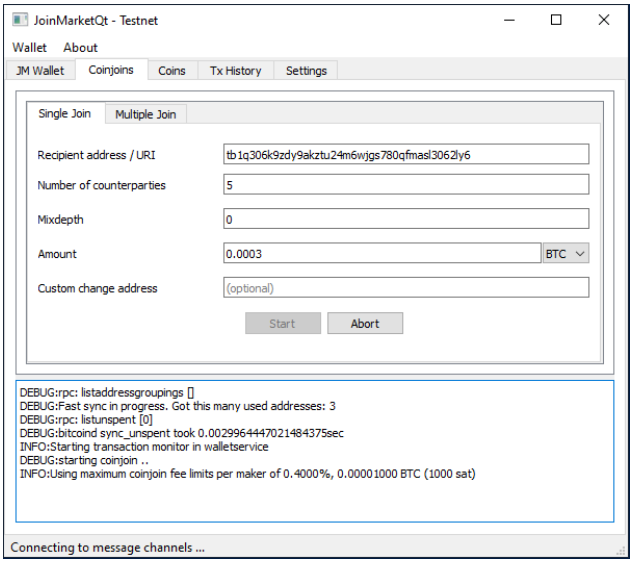


Fig. 6. JoinMarket CoinJoin.

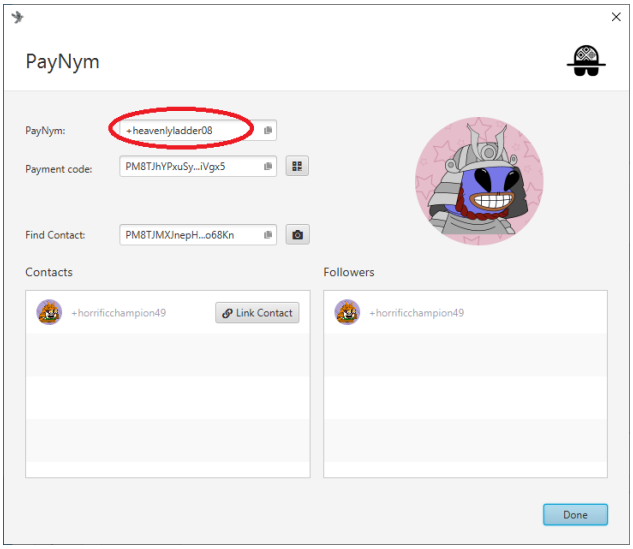


Fig. 7. Recipient PayNym identifier.

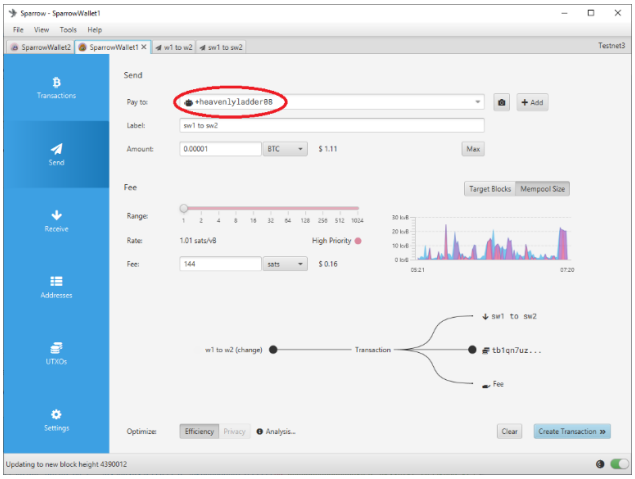


Fig. 8. PayNym imported in the sender wallet.

Step 5: Enable PayJoin in the recipient's Sparrow Wallet, allowing the recipient to contribute inputs and collaborate in constructing a mixed-input transaction.

Step 6: Complete the final transaction using both PayNym and PayJoin mechanisms, as depicted in Fig. 9.

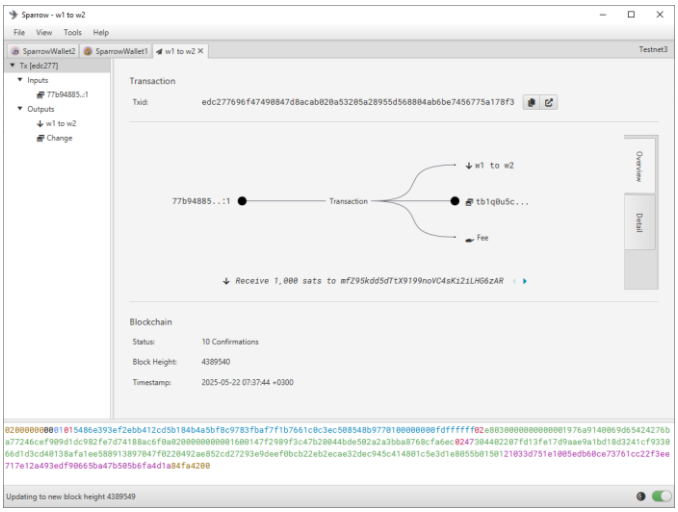


Fig. 9. Sparrow wallet transaction after broadcasting using PayNym and PayJoin.

Step 7: Ensure the Tor daemon is actively running so that all communications between wallets and the Bitcoin Core node are securely tunneled via the Tor network (Fig. 10).

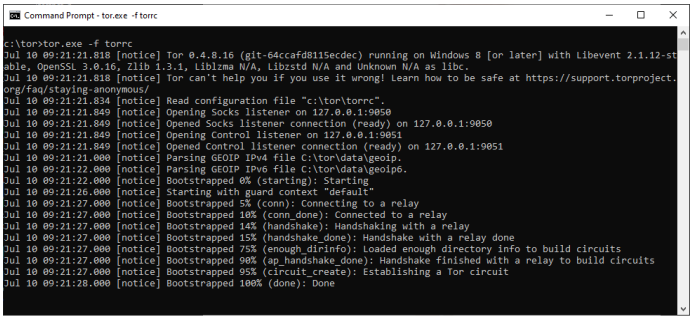


Fig. 10. Running Tor software.

IV. Experimental Results

This section presents the empirical outcomes resulting from the implementation of the proposed hybrid privacy-enhanced Bitcoin architecture within the Bitcoin Core testnet environment. The evaluation demonstrates how the integration of multiple privacy-preserving techniques enhances user anonymity, mitigates blockchain surveillance, and provides greater control over transactional privacy.

A. Integration of Privacy Tools

The system successfully integrated four core privacy-enhancing techniques—CoinJoin, PayJoin, Stealth Addresses, and Tor—into a cohesive framework built on open-source infrastructure. As summarized in Tables III and IV, testing conducted using Sparrow Wallet and JoinMarket confirmed that each component functioned correctly within the unified architecture.

- CoinJoin transactions were conducted using JoinMarket's maker-taker model, enabling collaborative transaction construction among multiple participants to expand the anonymity set.
- PayJoin transactions were successfully initiated and completed within Sparrow Wallet, effectively disrupting input ownership heuristics and demonstrating resilience against standard deanonymization techniques.
- Stealth Address functionality—leveraged via PayNym—enabled unlinkable transactions for the recipient, ensuring address-level anonymity and eliminating reuse-based traceability.
- Tor integration anonymized all network-level communications, safeguarding user IP addresses and metadata during wallet-to-node and peer-to-peer interactions.

TABLE III. Outcomes of Hybrid Privacy Model Layers

Layer	Technique	Benefit
Transaction origin obfuscation	CoinJoin (JoinMarket)	Breaks traceability by mixing funds, severing links to original funding sources (e.g., faucets)
Address unlinkability	PayNym	Prevents address reuse, making it infeasible to correlate payments to a static identifier
Input Ownership confusion	PayJoin (P2EP)	Combines the inputs of the sender and recipient, invalidating input ownership assumptions
Network anonymity	Tor proxy	Routes all traffic via Tor to obfuscate IP addresses and geographic metadata

TABLE IV. Evaluation Metrics for Hybrid Privacy Model Performance

Metric	Result	
Number of identifiable links	**Zero** – All transactions exhibited full unlinkability.	✓
Address reuse (PayNym)	**Eliminated** – New stealth addresses generated per transaction.	✓
Input ownership ambiguity (PayJoin)	**High** – Heuristics for input attribution rendered ineffective.	✓
CoinJoin anonymity set	**5 participants** – Sufficient for medium-strength mixing pools.	✓
Network/IP correlation	**None** – All traffic anonymized via onion routing.	✓

B. Drawbacks and Limitations

Although the proposed hybrid model substantially improves transactional privacy, it introduces several limitations and trade-offs that warrant consideration. These limitations primarily relate to usability, operational complexity, and system integration overhead.

CoinJoin's effectiveness relies on the active participation of both liquidity providers (makers) and takers; availability is not always guaranteed. In the testnet environment where user activity is limited, the anonymity set tends to be small. Consequently, CoinJoin rounds may experience latency, reduce efficiency and diminish practical usability for real-time scenarios.

C. Comparative Analysis

To contextualize the proposed framework, we compared it with recent studies (Table V) focusing on layered privacy, advanced stealth addresses, and mixing techniques. Key insights include:

- Privacy Strength – [24] demonstrate high address-level privacy, while [25] show moderate privacy gains using mixing techniques. Our framework combines multiple layers, achieving superior overall privacy.
- Resistance to Blockchain Heuristics – The hybrid model fully breaks input-output heuristics, outperforming the cryptography-focused approach in Springer [23] and the mixing-only approaches [25].
- Network-level Protection – Unlike the surveyed approaches, the proposed framework integrates Tor, providing robust network anonymity.
- Adoption and Usability – While more complex than single-technique solutions, the operational integration reduces coordination overhead compared to experimental or theoretical models.
- Implementation Feasibility – Unlike the mostly conceptual or prototype approaches in recent studies, the hybrid model is fully implemented and tested on the Bitcoin testnet.

Table V illustrates the comparison in detail, highlighting the hybrid framework's strengths relative to recent approaches.

TABLE V. COMPARISON OF PROPOSED HYBRID FRAMEWORK WITH RECENT PRIVACY APPROACHES

Framework / Approach	Privacy Strength	Resistance to Blockchain Heuristics	Network-level Protection	Adoption / Usability	Implementation Feasibility
Proposed Hybrid Framework	High – CoinJoin, PayJoin, Stealth Addresses, Tor	Strong – Fully breaks input-output heuristics	Full – Tor anonymization	Moderate – Integrated, some coordination required	High – Fully implemented on Bitcoin testnet
Mariani & Homoliak, 2025 [25]	Moderate – Mixing techniques	Moderate – Dependent on participant availability	Low – Transaction-level privacy only	Low–Moderate – Coordination overhead	Medium – Experimental
W. Zeming et al., 2024 [20]	High – Stealth addresses + note commitments	High – Conceals recipient identities	Low – No network-level anonymization	Low – Experimental	Medium – Prototype level
Springer survey, 2024 [23]	Moderate – Cryptography-based privacy	Moderate – Protects against basic heuristics	Low – Network-layer protection limited	Low – Implementation complex	Medium – Mostly theoretical
H. Schnoering & M. Vazirgiannis, 2023 [22]	Moderate – CoinJoin alone	Low–Moderate – Vulnerable to heuristics	Low – No network anonymity	Low – Limited adoption	Medium – Testnet / research implementations
R. Stütz et al., 2023 [5]	Moderate – Decentralized CoinJoin adoption	Low–Moderate – Actual privacy varies in practice	Low – No additional network protection	Moderate – Real-world adoption analyzed	Medium – Depends on wallet support
Yan et al., 2023 [24]	High – Homomorphic-encryption-enhanced stealth addresses	High – Effective against advanced analytics	Low – Network anonymity not included	Low – Technical adoption limited	Medium – Prototype

D. Discussion

The experimental results demonstrate that multi-layered hybrid architectures provide substantial improvements in privacy:

- Zero identifiable links were observed across all transactions.
- Address reuse was eliminated through dynamically generated stealth addresses.
- Input ownership heuristics were rendered ineffective, confirming the effectiveness of PayJoin.
- Tor integration ensured complete network-level anonymity.

Compared to recent studies [2–4], the hybrid framework provides both operational and theoretical advantages, integrating multiple privacy techniques in a single, practical implementation. The medium-sized CoinJoin anonymity sets and coordination requirements highlight ongoing challenges in balancing strong anonymity with usability, reinforcing the importance of continued research in user-friendly, multi-layer privacy solutions.

V. CONCLUSION AND FUTURE WORK

This study demonstrated that integrating CoinJoin, PayNym, PayJoin, and Tor within a unified hybrid wallet architecture can substantially enhance privacy in Bitcoin transactions. Each integrated technique contributes a distinct layer of protection: CoinJoin obscures transaction origin, PayJoin disrupts input ownership heuristics, PayNym ensures unlinkability of recipients, and Tor conceals network-level metadata. Our results show that the combination of these tools offers stronger anonymity than any single technique in isolation, increasing resistance to deanonymization heuristics, improving transaction unlinkability, and reinforcing overall resilience against blockchain surveillance.

The findings further highlight that a layered, multi-technique approach is essential in addressing the persistent arms race between privacy-preserving tools and blockchain forensic methods. In line with recent research emphasizing multi-layer strategies, our results confirm that hybrid models are not only

theoretically robust but also practically feasible when implemented through real-world wallets and testnet environments. At the same time, limitations remain: the operational complexity of coordinating different tools, the requirement for user technical proficiency, and compatibility constraints across wallet implementations. These challenges indicate that while enhanced privacy is achievable, usability and accessibility continue to be significant barriers to adoption.

Future research should prioritize enhancing the usability of the hybrid model by consolidating the various privacy tools into a unified, user-friendly wallet interface. Additional investigations on scalability, latency, and system performance under adversarial conditions on the Bitcoin mainnet are essential to assess real-world applicability. Future studies should also explore how different layers of privacy techniques interact under varying network loads, attack scenarios, and user adoption patterns. Broader comparative evaluations with alternative privacy-preserving strategies will help identify the trade-offs between anonymity strength, efficiency, and ease of use. Ultimately, the enduring challenge is to strike an effective balance between robust anonymity and practical usability, ensuring that advanced privacy solutions remain accessible to everyday Bitcoin users without introducing prohibitive complexity or performance limitations.

REFERENCES

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf.

[2] M. Conti, E. S. Kumar, C. Lal and S. Ruj, "A Survey on Security and Privacy Issues of Bitcoin," *IEEE Communications Surveys & Tutorials*, 2018.

[3] N. Amarasinghe, X. Boyen and M. McKague, "A Survey of Anonymity of Cryptocurrencies," in *ACM*, 2019.

[4] G. Fanti and P. Viswanath, "Deanonymization in the Bitcoin P2P Network," in *ACM Conference on Computer and Communications Security (CCS)*, 2017.

[5] R. Stütz, J. Stockinger, P. Moreno-Sanchez, B. Haslhofer and M. Maffei, "Adoption and Actual Privacy of Decentralized CoinJoin Implementations in Bitcoin," in *4th ACM Conference on Advances in Financial Technologies*, 2023.

[6] S. Ghesmati, A. Kern, A. Judmayer and E. Weippl, "Unnecessary Input Heuristics and PayJoin Transactions," in *HCI International 2021 - Posters*, 2021.

[7] T. B. Manual, "What Is A Bitcoin PayNym?," 2022. [Online]. Available: https://thebitcoinmanual.com/articles/bitcoin-paynyms/.

[8] J. Cui, C. Huang, H. Meng and R. Wei, "Tor network anonymity evaluation based on node anonymity," *Cybersecurity*, vol. 6, 2023.

[9] A. Wahrstätter, A. Taudes and D. Svetinovic, "Reducing Privacy of CoinJoin Transactions: Quantitative Bitcoin Network Analysis," *IEEE Transactions on Dependable and Secure Computing*, vol. 21, no. 5, p. 4543–4558, 2024.

[10] M. C. K. Khalilov and A. Levi, "A Survey on Anonymity and Privacy in Bitcoin-Like Digital Cash Systems," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, 2018.

[11] S. Ghesmati, W. Fdhila and E. Weippl, "SoK: How private is Bitcoin? Classification and Evaluation of Bitcoin Privacy Techniques," in *International Conference on Availability, Reliability and Security*, Vienna, Austria, 2022.

[12] G. Maxwell, "Coinjoin: Bitcoin privacy for the real world," *bitcointalk*, 2013.

[13] J. Fan, Z. Wang, Y. Luo, J. Bai, Y. Li and Y. Hao, "A New Stealth Address Scheme for Blockchain," in *ACM Turing Celebration Conference*, China, 2019.

[14] B. Project, "What Is A Full Node?," 2025. [Online]. Available: https://bitcoin.org/en/full-node#what-is-a-full-node.

[15] "Bitcoin Core," [Online]. Available: https://bitcoin.org/en/bitcoin-core/.

[16] S. Ghesmati, W. Fdhila and E. Weippl, "Usability of Cryptocurrency Wallets Providing CoinJoin Transactions," *IACR Cryptology ePrint Archive*, 2022.

[17] S. Houy, P. Schmid and A. Bartel, "Security Aspects of Cryptocurrency Wallets – A Systematic Literature Review," *ACM Computing Surveys*, vol. 56, no. 1, pp. 1-31, 2023.

[18] "JoinMarket: Decentralized Bitcoin CoinJoin implementation," github repository, 2024. [Online]. Available: https://github.com/JoinMarket-Org/joinmarket-clientserver.

[19] "Sparrow Wallet," [Online]. Available: https://sparrowwallet.com/.

[20] W. Zeming, F. Jiawen, H. Zhicheng, Z. Yu, M. Shansi, Z. Junlang, L. Chufeng, Z. Gansen and T. Hua, "Privacy Protection Method for Blockchain Transactions Based on the Stealth Address and the Note Mechanism," *Applied Sciences*, vol. 14, no. 4, 2024.

[21] G. Fanti, S. B. Venkatakrishnan, S. Bakshi, B. Denby, S. Bhargava, A. Miller and P. Viswanath, "Dandelion++: Lightweight Cryptocurrency Networking with Formal Anonymity Guarantee," in *ACM Measurement and Analysis of Computing Systems*, 2018.

[22] H. Schnoering and M. Vazirgiannis, "Heuristics for Detecting CoinJoin Transactions on the Bitcoin Blockchain," arXiv preprint, 2023.

[23] X. Luo, X. Chen, X. Chen and Q. Cheng, "A survey on the application of blockchain in cryptographic protocols," *Cybersecurity*, vol. 79, no. 7, 2024.

[24] Y. Yan, G. Shao, D. Song, M. Song and Y. Jin, "HE-DKSAP: Privacy-Preserving Stealth Address Protocol via Additively Homomorphic Encryption," *arXiv*, 2023.

[25] J. Mariani and I. Homoliak, "SoK: A Survey of Mixing Techniques and Mixers for Cryptocurrencies," *arXiv*, 2025.

[26] "Blockstream Explorer," [Online]. Available: https://blockstream.info/testnet/. [Accessed July 2025].

[27] "Mempool," [Online]. Available: https://mempool.space/testnet. [Accessed July 2025].

[28] "Wireshark," [Online]. Available: https://www.wireshark.org/.

[29] CoinFaucet.eu, "Bitcoin Testnet Faucet," 2025. [Online]. Available: https://coinfaucet.eu/en/btc-testnet/.

VGG-19 and Vision Transformer Enabled Shelf-Life Prediction Model for Intelligent Monitoring and Minimization of Food Waste in Culinary Inventories

Bindhya Thomas, Dr Priyanka Surendran
College of Computer Studies, University of Technology, Bahrain

***Abstract*—Food waste, particularly in the prepared food industry, presents a serious worldwide concern with serious ethical, environmental and socioeconomic implications. In restaurants and catering contexts, traditional inventory and waste management systems frequently lack the versatility and granularity to mitigate spoilage in real-time. The study proposes a sophisticated deep learning framework that predicts the remaining shelf-life of prepared food items using visual input, enabling timely interventions to reduce food waste. The proposed hybrid architecture integrates VGG-19 (Visual Geometry Group 19-layer network) for fine-grained feature extraction with Vision Transformer (ViT) that models contextual degradation patterns and temporal cues. The model operates by analyzing food images at regular intervals and predicting the remaining time before spoilage, enabling proactive decision-making for consumption prioritization. Food images are categorized into four freshness states: Fresh, Fit for Consumption, About to Expire and Expired, enabling the model to monitor real-time conditions. An elaborate dataset with 34 distinct food categories was utilized in the study, achieving outstanding performance with 98% accuracy, 97.5% precision, 97.9% recall and an F1-score of 97.75% and yielded an estimated 84% reduction in food waste. The model stands out for its non-invasive, image-based decision-making and the potential scalability across various food service settings. By offering predictive insights into food degradation and by using only visual data, the study advances the integration of artificial intelligence into sustainable food management.**

Keywords*—Food waste reduction; shelf-life prediction; VGG-19; vision transformer; image-based freshness classification; sustainable food management

I. Introduction

Food waste represents one of the most critical and paradoxical challenges of the modern world, with a gigantic volume of edible resources being discarded while millions continue to experience food insecurity and hunger [1]. It is estimated that around one-third of all food produced, amounting to over 1.3 billion tons annually, is wasted across the supply chain, from farms and distribution hubs to retail outlets and households [2]. Production and logistics inefficiencies, overstocking, unpredictable consumer behavior, inadequate preservation technologies and poor inventory management are some of the many factors contributing to food waste generation [3], [4]. Food waste reduction is a vital social, environmental and ethical necessity that goes beyond efficiency. It is becoming more and more critical to reduce waste through more intelligent, flexible solutions, as the world's population continues to expand and the effects of climate change put more strain on food systems [5]. Procedural and policy-based strategies like enhanced inventory checks, manual expiration date monitoring, redistributive networks and consumer awareness campaigns were traditionally adopted to curb food waste generation [6]. These methods mostly depend on human judgement and intervention, making them vulnerable to frequent errors and overlooks. The product-level data and real-time environmental variables are generally ignored, relying singularly on human visual inspection to judge freshness, which is inappropriate and inefficient for a vast amount of food. Similarly, demand estimates in the catering and hospitality sectors are frequently based on historical patterns and managerial judgement that often results in overproduction and associated food wastage [7]. The absence of traceability and consistent monitoring in retail and agricultural supply chains leads to spoilage that could have been avoided. Despite being fundamental, these conventional approaches are ineffective at handling the complexity of contemporary food systems as they are reactive rather than proactive.

A revolutionary change in handling food waste was promised by the rise of data-driven technologies. Predictive analytics, real-time monitoring and intelligent inventory control were made possible by blockchain, artificial intelligence (AI), computer vision and Internet of Things (IoT) sensors [8]. Convolutional neural networks (CNNs), YOLO object detection and recurrent neural networks (RNNs) are examples of machine learning (ML) models that have shown great promise in detecting food products, forecasting demand and identifying spoilage [9]. Automated supply chain optimizers, dynamic pricing systems and smart refrigeration systems have all been tested in business and industrial settings. Despite the initial potential, high implementation costs, reliance on data, lack of interoperability and restricted adaptability in rural or low-resource environments hamper their adaptability, generalizability and scalability. The systemic efficiency is undermined by the siloed nature of most of the current solutions, which concentrate on discrete supply chain components.

The central research question addressed in this study is whether a hybrid deep learning framework that integrates convolutional neural networks and transformer-based self-attention mechanisms using only visual inputs can reliably predict the remaining shelf life of prepared food items, enable timely interventions and significantly reduce food waste in real-world service settings. This study proposes a novel hybrid architecture for remaining shelf-life prediction, combining the

strengths of deep learning (DL) with advanced image segmentation and self-attention mechanisms. By using visual cues from food items to infer freshness levels, the study enables more precise and automated decisions on redistribution, consumption prioritization and dynamic pricing. The primary objectives of the study are as follows:

- Develop a CNN-Transformer hybrid DL architecture for accurate classification of food items into different freshness categories based on visual cues extracted from prepared food images.
- Evaluate the model performance and compare it with conventional methods, demonstrating its practical impact through a measurable reduction in food waste by enabling timely consumption or disposal based on predicted freshness.

The further sections of the study are structured as follows: Section II offers a detailed analysis of the latest advances in food waste reduction research and highlights the current research constraints. Section III delineates the proposed methodology. Section IV presents the experimental findings accompanied by a comprehensive analysis of the model performance in Section V. Finally, Section VI concludes the study by encapsulating the principal findings and emphasizing prospective areas for further research.

II. Related Works

Chun et al. [10] suggested deep learning techniques in food image classification in Korean cuisine. The study employed a Korean food image dataset that contained diverse categories of dishes that were pre-processed and augmented to enhance model generalization. Several convolutional neural network architectures were tested through transfer learning, among which InceptionResNetV2 achieved the highest classification accuracy of 81.91%, though it required an extensive training time of 436,182 seconds due to its complex inception–residual structure. NasNetLarge and MobileNetV2 followed with accuracies of 77.91% and 75.36%, respectively. Traditional ResNet variants, including ResNet-101V2, ResNet-152V2, and ResNet-50V2, showed lower accuracies ranging from 73.7% to 68.27%. The computational intensity was a major limitation of the study.

Louro et al. [11] suggested convolutional neural network (CNN)-based approaches for food waste reduction through food recognition technology and promote sustainable eating practices. The study utilized Food-101 dataset, together with the ResNet-50 initially and later adopted transfer learning with ImageNet weights to avoid overfitting. The convolutional layers processed food images hierarchically, progressively learning low-level features such as edges and textures before combining them into higher-level representations of ingredients and dishes and achieved a 90% classification accuracy. However, the model was limited by its computational cost and training time, that hindered scalability and real-time deployment in resource-constrained environments.

Dey et al. [12] proposed SmartNoshWaste, a blockchain-based multi-layered system to reduce food waste within the Farm-to-Fork supply chain. Data System Architecture comprising blockchain technology, QR codes and cloud computing to digitize and store food-related data and the ML module employing Q-learning-based reinforcement learning to optimize decisions formed the two basic blocks of the framework. Production, processing, distribution, retailing and consumption were the five main supply chain phases that were functionally tracked and examined. Real-world potato waste data from the nosh app was used for experimental evaluations, and a 9.46% decrease in food waste compared to the baseline data was observed. The dependence on a single food item and context-specific evaluation limited the generalizability and scalability to larger agricultural or urban food systems.

Nascimento et al. [13] developed an AI-driven model for reducing food waste through improved production planning of own-branded products in grocery stores in Brazil. Using historical daily sales data for a year, the study compared five ML algorithms: logistic regression (LR), multilayer perceptron (MLP), DT (J48), PART and random forest (RF). The algorithms forecasted revenue levels, convertible into product demand and rule-based models and RF outperformed others with 90.17% accuracy. A considerable reduction in total food waste across 312 days was observed: from 6,169 kg under aggressive strategies and 653 kg under balanced strategies to just 117.64 kg, representing an 82% reduction. The insights were limited, as the focus was on a single grocery store, restricting the applicability to broader retail settings or more complex supply chains.

Rasyidi et al. [14] aimed to create an elderly-friendly food recording application in Indonesia by overcoming the shortcomings of existing datasets and models that failed to capture the complexity of local cuisine. The study introduced a new food dataset of 24,427 images covering 160 Indonesian food categories and evaluated 67 models built on 16 state-of-the-art deep learning architectures. EfficientNet V2L achieved the best performance with an accuracy of 85.44% and a top-5 accuracy of 97.84%, outperforming models like ConvNeXt Large and Swin-S. The study was, however, limited by difficulties with single-label classification, variations in food presentation, and complex image compositions that hampered generalization across real-world food recognition scenarios.

Jacob et al. [15] utilized AI-driven optimization for food waste reduction in the cassava processing supply chain to transform farm-to-table operations for Garri production. The study utilized data collected from 4,200 respondents across seven states and 42 local government areas, with demand forecasting and inventory optimization performed by the AI framework incorporating regression analysis and DT algorithms, while natural language processing (NLP) analyzed qualitative interview data to extract stakeholder insights. The framework utilized data mining in detecting inefficiencies and waste points across the supply chain, enabling targeted interventions. A 40% reduction in food waste, 30% decrease in processing time and a 25% cut in transportation costs were observed due to optimized logistics and production scheduling. Scalability was a concern as implementation phase was hindered by poor data quality and inadequate infrastructure affecting the consistency of the AI deployment across the supply chain.

Hübner et al. [16] conducted a life cycle assessment (LCA) of "Foodforecast", a cloud-based ML service reducing bakery food waste comprising sales forecasting, cloud computing infrastructure and hardware resources. Historical sales data was utilized to generate optimized forecasts via ML algorithms to minimise overproduction and quantifying environmental impacts through life cycle modelling. The study evaluated four environmental impact categories: global warming, abiotic resource depletion, cumulative energy demand and freshwater eutrophication and achieved an average 30% reduction in bakery returns, equating to a total decrease of 2000 tons of waste on evaluation in real-world data from 175 bakeries. However, the methodological uncertainties and inadequate data constrained the generalizability of the study.

Sigala et al. [17] suggested a fully automated AI-based food waste tracking system deployed across various HORECA (Hotel + Restaurant + Cafe/Catering) establishments in Europe. The study integrated a hardware unit comprising a scale and an IoT-enabled camera device, positioned beneath and above food waste bins, respectively. The system detected each waste event by recording the weight of discarded food and simultaneously captured images, which were transmitted to a cloud-based system. The advanced image recognition and DL algorithms detected and segregated food waste into groups, specific items and categories of avoidability and source, enabling tailored operational changes, including portion control, menu redesign and improved food storage, achieving a 23 to 51% reduction in food waste across most sites. The absence of a detailed cost analysis hampered the study.

To improve inventory oversight and minimize food wastage in supermarkets, Li et al. [18] proposed a real-time inventory tracking framework leveraging computer vision techniques. YOLOv5 object detection algorithm formed the core of the method, that utilizes CNN to autonomously detect and enumerate items in shelf images submitted by users. These images are analyzed to extract item data, which is then synchronized with a Firebase database, while the front-end is dynamically updated using a Flutter-based application. The system achieved an image processing time of approximately 0.79 seconds, with a precision of around 50% and a recall rate of 80%. The accuracy declined in scenarios involving densely packed or multi-layered shelf views, suboptimal image angles, and a limited recognition vocabulary led to frequent miscounts and missed detections.

Min et al. [19] proposed a Stacked Global-Local Attention Network (SGLANet) for food recognition by capturing both global and local discriminative features of food images. The study employed the ISIA Food-500 dataset, comprising 399,726 images across 500 diverse categories. The architecture consisted of two complementary subnetworks: one applied hybrid spatial-channel attention to learn global-level characteristics such as texture and shape, while the other leveraged cascaded spatial transformers to identify ingredient-relevant local regions and aggregate regional information. By fusing these global and local representations, SGLANet produced a more comprehensive feature space for classification. The model achieved strong performance, recording a validation accuracy of 90.92%. However, the computational complexity and training time, made the model less practical for real-time deployment in lightweight environments such as mobile devices.

To curtail food waste within agricultural supply chains, Wang [20] introduced an AI-enhanced Decision Support System (DSS) for intelligent inventory management and optimized resource allocation. The system architecture consisted of three integral layers: real-time data acquisition, an AI-driven decision-making core and adaptive learning through continuous feedback loops. To forecast demand, NNs and support vector machines (SVM) were employed, achieving predictive accuracies of 90% and 85%, respectively. Inventory distribution across the supply chain was optimized using heuristic algorithms, including genetic algorithms (GA) and particle swarm optimization (PSO), which reduced spoilage by 20% and 22%, respectively. High dependency on uninterrupted, high-quality data streams presented challenges for deployment in rural or low-resource agricultural environments.

Rodrigues et al. [21] compared four ML models in forecasting demand within food catering services to minimize food waste arising from overproduction or underestimation. The study employed RF, LightGBM, Long Short-Term Memory (LSTM) networks and Transformer-based models. Two baseline approaches, a naïve model and a moving average method, were developed to simulate conventional forecasting practices and served as benchmarks across three distinct food service settings. The RF algorithm delivered the most accurate predictions in two of the cases, whereas the LSTM model demonstrated superior performance in the third, collectively contributing to food waste reductions ranging from 14% to 52%, and lowering unmet demand by up to 16% relative to the baselines. The study's practical scope was constrained as the focus was on a single dish, restricting applicability in more varied or complex menu environments.

Goh & Yann [22] suggested an Inception-V3 model with transfer learning for food image classification using the Food-101 dataset. The model leveraged convolutional layers for extracting deep spatial features, while transfer learning from ImageNet enhanced its recognition capability. The architecture operated by factorizing convolutions into smaller kernels and incorporating auxiliary classifiers, while retaining depth for complex feature extraction. The transfer learning setup adapted pre-trained ImageNet weights to food-specific features, enabling faster convergence and more accurate recognition. The study achieved an accuracy of 90%, but was limited by the intra-class variability and noise within the dataset, that reduced classification stability.

Faezirad et al. [23] proposed a ML-based forecasting approach to reduce food waste in an Iranian university dining systems by predicting student attendance using both reservation and behavioral data and identified fluctuating student presence as the key contributor in food surplus. The two-stage prediction framework employed an artificial neural network (ANN) to model both deterministic and stochastic aspects of demand. In the first stage, the ANN generated a point estimate for student turnout and the system optimized the total operational cost, balancing the trade-off between food wastage and shortage penalties in the second stage. The model achieved accuracy rates between 73% and 75%, resulting in a reported 79.66% reduction

in food waste over a one-year period. The dependence on historical behavioral data from a single institution limited the model's adaptability to other educational environments.

Malefors et al. [24] proposed ML-based forecasting models to anticipate guest attendance in public catering facilities, for optimizing meal preparation and minimizing food waste under COVID-19 pandemic conditions. Attendance data from 18 primary schools and 16 preschool kitchens across Sweden, both pre- and during-pandemic periods, was collected and evaluated using several AI techniques, including ANN, Poisson autoregressive models and RF algorithms. RF model demonstrated superior accuracy with a conditional mean absolute error below 0.15 for training data and between 0.448 and 0.487 for kitchen-level forecasts. The implementation of forecasting yielded financial savings estimated between €921 and €1,298 compared to operations without predictive support. The models exhibited limited reliability during the initial phase of the pandemic and were challenged by abrupt surpluses, indicating reduced robustness under extreme uncertainty.

A. Research Gap

Despite the progress made in food image recognition and subsequent food waste recognition using deep learning architectures such as ResNet, InceptionResNetV2, MobileNetV2, EfficientNet and SGLANet on large-scale image datasets, these studies have primarily addressed the challenge of food categorization rather than the more critical issue of food waste reduction [11] [19] [22]. In parallel, approaches based on predictive analytics, regression models, neural networks, decision trees, reinforcement learning and optimization techniques, including genetic algorithms, random forests, LSTM networks, and Transformer-based forecasting have been widely applied for inventory management, demand forecasting and supply chain optimization [13] [15] [21]. These models have demonstrated measurable reductions in waste but are fundamentally dependent on structured transactional or sensor-based data streams, overlooking the immediate and non-invasive potential of image-based monitoring. This divergence highlights a critical research gap: existing studies either focus on food recognition without waste-oriented outcomes or address waste reduction without leveraging visual cues. Thus, there is a pressing need for an integrated framework with freshness-annotated image data to deliver accurate shelf-life prediction and provide actionable strategies for minimizing food waste in real-world service operations.

III. Materials and Methods

The proposed hybrid model utilizes image data to precisely estimate the remaining shelf-life of prepared foods by combining convolutional and transformer-based architectures. The study makes use of Food Image Classification dataset that has undergone rigorous preprocessing and augmentation and an average shelf-life table is employed for reference in the training stage. Rich spatial characteristics are extracted from input images by the feature extractor, VGG-19, and then used by the Vision Transformer (ViT) to represent temporal and contextual dependencies for accurate shelf-life prediction. Fig. 1 illustrates the basic architecture of proposed model.

A. Dataset Description

The dataset utilised in this study is the Food Image Classification Dataset [25], from a publicly accessible Kaggle repository, comprising around 24,000 high-resolution images spanning 34 diverse food categories, covering a wide range of both Indian and Western cuisines, representing a realistic mix of freshly prepared and commonly consumed dishes. The images were collected from various sources and curated to ensure diversity in lighting conditions, presentation styles and angles, with each food image labelled according to its class. In restaurant and catering inventory analysis, where food items are prepared in bulk and require regular monitoring, the dataset serves as an essential analytical tool. The construction of sophisticated structures, expiration trends and utilisation strategy optimisation are made possible by the visual diversity and class annotations. This makes the dataset perfect for food service operations' waste minimisation and real-time shelf-life estimate algorithms. Fig. 2 represents the sample images in the dataset over different food categories.

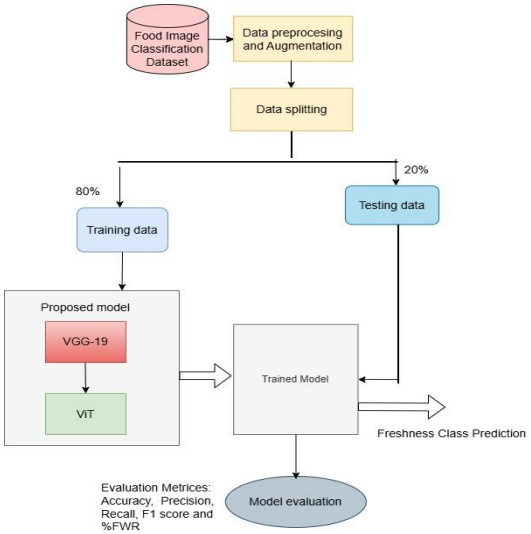


Fig. 1. Basic architecture of the proposed model.



Fig. 2. Sample images in the food image classification dataset.

B. Exploratory Data Analysis

In order to find patterns, anomalies and underlying structures in the dataset, exploratory data analysis or EDA, is an essential step. EDA enables to assess the quality and variability of the images, identify class imbalances and evaluate the distribution of classes in image classification tasks. Additionally, it offers information on possible redundancy, noise or labelling irregularities that have the potential to impair model performance. EDA guarantees well-informed judgments being taken prior to preprocessing and model building by visualizing and encapsulating the data. Fig. 3 illustrates the distribution of images across the 34 food categories in the dataset. Certain subgroups, such as Baked Potato, Hot Dog, Donut and Crispy Chicken, are well-represented, while others, such as Paani puri, Samosa and Kulfi, have relatively fewer samples. Addressing the imbalance is of vital importance as it may bring about in biased learning when the model is being trained. By using EDA to identify this distribution, balanced data augmentation and sampling techniques can be developed to enhance model generalization.

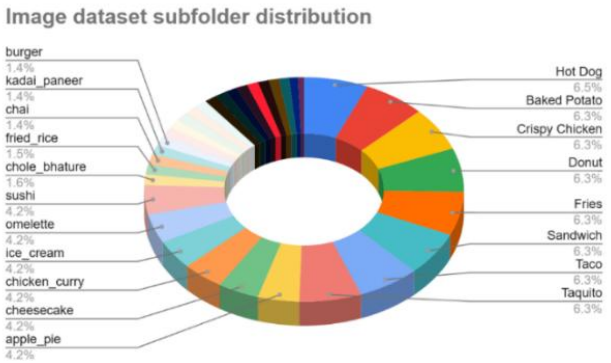


Fig. 3. Image dataset subfolder distribution.

C. Data Preprocessing and Augmentation

A structured data preprocessing and augmentation pipeline is designed for the Food Image Classification dataset to ensure clear and standardized input for the subsequent hybrid architecture. The stage ensures uniformity in image dimensions, address class imbalance and enrich the training set with diverse transformations to reduce overfitting and a compatible data input for the proposed models is generated.

The images are first resized to a fixed dimension of 224×224 pixels, as shown in Eq. (1), conforming to the input requirements of the VGG-19 architecture.

$$I_{resized} = Resize(I_{original}, (H, W)) \quad (1)$$

where, $I_{original}$ is the raw image and $H, W = 224$, the target height and width. Image normalization is further applied to scale pixel intensity values to a standard range, to stabilize the training process and accelerate convergence. The pixel values are normalized to the range [0, 1], as shown in Eq. (2):

$$I_{norm} = \frac{I_{resized}}{255} \quad (2)$$

As the pretrained weights are leveraged, mean subtraction and division by standard deviation were performed as per Eq. (3):

$$I_{norm} = \frac{I_{resized} - \mu}{\sigma} \quad (3)$$

where, μ and σ are the channel-wise mean and standard deviation, respectively. This standardization ensures that each input image contributes uniformly during gradient updates, reducing internal covariate shift. To enable supervised classification, the images are labelled not only by food category but also by freshness state, using a four-class scheme: Fresh, Fit for Consumption, About to Expire and Expired. To numerically represent these classes, label encoding was applied, assigning each category a unique integer value to ensure compatibility with the proposed hybrid model.

Let $C = \{C_0, C_1, C_2, C_3\}$ be the set of freshness categories, where C_0, C_1, C_2, C_3 represent Fresh, Fit for Consumption, About to Expire and Expired, respectively. Each image is tagged based on its freshness level determined by the food's time since preparation t and its standard shelf-life x, as shown in Eq. (4):

$$y_i = \begin{cases} 0\ ;if\ t < 0.2x\ (fresh) \\ 1\ ;\ if\ 0.2x \le t \le 0.8x\ (Fit\ for\ consumption) \\ 2\ ;\ If\ 0.8x < t < x\ (About\ to\ expire) \\ 3\ ;\ If\ t \ge x\ (Expired) \end{cases} \quad (4)$$

The encoded label $yi \in \{0,1,2,3\}$ allows the model to learn the visual differences associated with freshness states, thereby supporting classification tasks aligned with expiration forecasting. Table I illustrates the average shelf-life of various food items.

TABLE I. AVERAGE SHELF LIFE OF VARIOUS FOOD ITEMS

Food Item	Room Temp (hours)	Refrigerated (days)	Frozen (months)
Hot Dog	2	7	1-2
Baked Potato	2	3-5	Not Recommended
Crispy Chicken	2	3-4	4
Donut	24-48	5-7	2-3
Fries	2	2-3	1
Sandwich	3	3	1-2
Taco	2	2-3	1-2
Taquito	2	3-4	2
Apple pie	48	4-5	6-8
Cheese cake	2	5-7	2
Chicken curry	6	3-4	2-3
Ice cream	N/A	30-60	2-4
Omelet	2	2-3	Not Recommended
Sushi	2	1	Not Recommended
Chole bhature	2	2-3	1
Fried rice	6	3-4	1
Tea (Chai)	1	1-2	Not Recommended
Kadai paneer	2	3-4	2
Burger	2	2-3	1
Chapati	24	3-	1
Momos	2	2-3	1-2

Butter naan	24	3-4	1
Pav bhaji	2	2-3	1-2
Idli	24	3-4	1
Dal makhani	4	4-5	2-3
Jalebi	48-72	7	2
Kaathi rolls	24	2-3	1
Pizza	2	3-4	1-2
Masala dosa	2	2-3	Not Recommended
Pakode	2	2-3	1
Dhokla	24	3-4	1-2
Samosa	8	4-5	1
Kulfi	N/A	7-14	2-3
Paani puri	2	1-2	Not Recommended

The dataset is now partitioned into training and testing subsets using an 80:20 split in a randomized fashion. This stratification ensures that the model learns from a broad spectrum of data, while being evaluated on unseen instances. Overall class distribution is maintained to avoid sampling bias and class imbalance. As dataset exhibited class imbalance identified through EDA, the potential of this disparity to create bias during training hampering generalization is considered. Class weights are calculated and added to the loss function to counteract this, allowing the model to penalize minority class misclassification more severely. The weight w_c assigned to class c is determined as in Eq. (5):

$$w_c = \frac{N}{n_c \times C} \tag{5}$$

where, N is the total number of samples and n_c, the number of samples in class c and C is the total number of classes, the four freshness classes. These weights are passed to the training loop so that the model learns more effectively from all classes regardless of frequency.

Data augmentation is applied to the training images to simulate real-world variability in food images, such as differences in angle, lighting and scale, enabling the model to learn invariant features. It includes a number of operations, including random horizontal and vertical flipping, slight rotation, width and height shifts, zooming and brightness variation and the augmented image I' is obtained from the original image I, as shown in Eq. (6):

$$I' = T(I) = R(\theta) \circ Z(s) \circ S(x, y) \circ F(I) \tag{6}$$

where, $R(\theta)$, $Z(s)$ and F denote rotation by angle θ, zooming by scale factor s and flipping transformations, respectively, and $S(x, y)$ represents a shift in width and height. The augmentation steps are applied only to the training dataset to avoid data leakage into test sets. After augmentation, the training dataset increased from 19,200 images to approximately 38,400 images, resulting in a total dataset size of about 43,200 images. By synthetically expanding the dataset, the model is better equipped to handle diverse food presentations and mitigate overfitting.

The images are further converted into tensor format to be efficiently processed by the hybrid model and also to support GPU acceleration. For optimal speed and generalization, the dataset is further segmented into smaller batches, enabling the model to handle multiple samples at once. Prefetching lowers data loading latency and offers more seamless training cycles by preparing the subsequent batch while the current one is being processed.

D. Model Development

1) VGG-19 (Visual Geometry Group-19): VGG-19 is a deep CNN developed to perform robust image classification tasks by learning hierarchical representations of visual data [26]. The VGG-19 network model depicted in Fig. 4, showcases a deep CNN architecture consisting of 19 weight layers. The input image with dimensions 224×224×3 is initially passed through multiple stacked convolutional layers employing small 3×3 kernels with a stride of 1 and padding, followed by ReLU activation functions to introduce non-linearity.

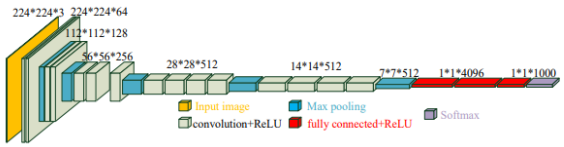


Fig. 4. VGG-19 network model architecture.

As the data progresses through the network, the number of feature maps increases from 64 to 512, while spatial dimensions reduce due to the interleaved max pooling layers which halve the resolution using 2×2 windows. Following the last convolutional block, the flattened feature maps are fed into three fully connected layers, two of which are 4096 in size and one final layer, which is 1000 in size. All of these layers are triggered by ReLU, with the exception of the last one, which outputs class probabilities using Softmax. This particular layer is excluded in the proposed hybrid model. By repeating tiny filters, this design highlights depth and aids in capturing hierarchical representations and fine-grained information. Each convolution layer performs feature extraction, as shown in Eq. (7):

$$X_l = \sigma(W_l * X_{l-1} + b_l) \tag{7}$$

where, X_{l-1} is the input feature map to layer l and W_l, b_l are trainable weights and biases of l. σ and $*$ denotes the ReLU activation function and convolution operation, respectively. The change in feature map size after convolution can be expressed as in Eq. (8) and Eq. (9):

$$H_{out} = \left\lfloor \frac{H_{in} - K + 2P}{S} \right\rfloor + 1 \tag{8}$$

$$W_{out} = \left\lfloor \frac{W_{in} - K + 2P}{S} \right\rfloor + 1 \tag{9}$$

where, H_{in} and W_{in} are the input height and width, respectively. K, P and S represent the kernel size, padding and stride, respectively. The max-pooling operation reduces spatial dimensions while preserving critical information and is defined as in Eq. (10):

$$X_{pooled}(i, j) = \max_{(m,n) \in \Omega} X(i + m, j + n) \tag{10}$$

where, Ω represents the pooling window, (i, j) are the top-left coordinates of the pooling window and m, n are the indices over the local region. The fully connected operation that follows convolution and pooling is defined as in Eq. (11):

$$z = W_{fc} \cdot x + b_{fc} \tag{11}$$

where, x is the flattened input feature vector and z is the linear transformation output. For the proposed hybrid model, the final classification layer is removed, and the intermediate feature representation is fed into downstream models for task-specific inference. The final feature maps are flattened and passed through fully connected layers, producing a high-level embedding.

2) Vision Transformer (ViT): ViT adapts the self-attention-based architecture of transformers from NLP to computer vision tasks by processing images as sequences of patches instead of relying on convolutional filters [27]. It segments an image into fixed-size patches, embeds positional information in these patches and uses self-attention mechanisms to model global contextual relationships throughout the visual field. ViT achieves better performance in applications requiring holistic image processing by capturing long-range relationships through stacked transformer layers, in contrast to convolutional networks that rely on local receptive fields. Fig. 5 illustrates the architecture of ViT.

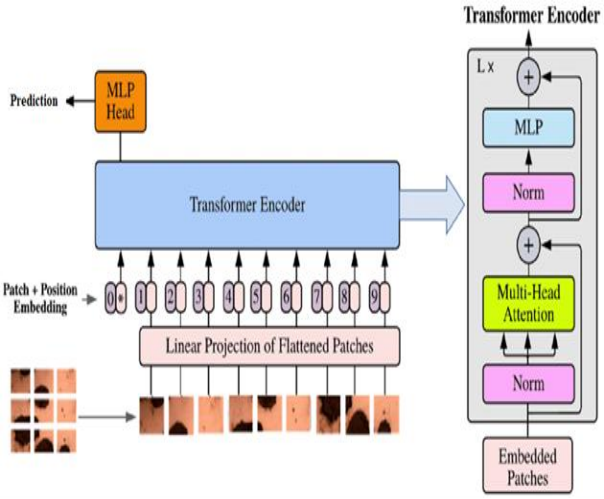


Fig. 5. Basic architecture of ViT.

The process begins with patch embedding, where the input image $X \in \mathbb{R}^{H \times W \times C}$ is divided into $N = \frac{HW}{P^2}$ non-overlapping patches of size $P \times P$. Each patch is then flattened and projected into a latent vector using a trainable linear projection, as shown in Eq. (12):

$$z_0^i = x^i \cdot E \; ; for \; i = 1,2, \dots, N \tag{12}$$

where, $x^i \in \mathbb{R}^{P^2 \cdot C}$ is the i-th patch and $E \in \mathbb{R}^{(P^2 \cdot C) \times D}$ is the embedding matrix. Spatial information lost in flattening is retained by adding positional embeddings $E_{pos} \in \mathbb{R}^{(N+1) \times D}$ to the patch embeddings, including a special learnable CLS token used for classification, as shown in Eq. (13):

$$z_0 = [x_{cls}; x^1 E; x^2 E; \dots ; x^N E] + E_{pos} \tag{13}$$

The embedded sequence is then passed through L identical transformer encoder blocks, each comprising a Multi-Head Self-Attention (MSA) layer and a Multi-Layer Perceptron (MLP) block, both followed by Layer Norm and residual connections, as shown in Eq. (14) and Eq. (15):

$$z_l' = MSA(LN(z_{l-1})) + z_{l-1} \tag{14}$$

$$z_l = MLP(LN(z_l')) + z_l' \tag{15}$$

where, $l = 1,2, \dots, L$ and LN denotes Layer Normalization. The core of ViT lies in self-attention, that models dependencies across all patches. For each attention head, the attention scores are computed, as shown in Eq. (16):

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{16}$$

where, $Q = zW^Q, the \; query$, $K = zW^K$, the key and $V = zW^V$, the value projection. d_k represents the dimension of the key vectors. The output corresponding to the CLS token after the final transformer block is passed through an MLP head to produce the final class probabilities, as shown in Eq. (17):

$$y = softmax\ (W_{head} \cdot z_L^{(cls)}) \tag{17}$$

3) Proposed VGG 19-ViT hybrid model: The proposed hybrid model integrates the robust spatial feature extraction capabilities of VGG-19 with the temporal reasoning and attention mechanisms of ViT to effectively classify food items based on their visual freshness condition. The VGG-19 network processes the input image by passing it through a deep convolutional stack, extracting hierarchical spatial features from local edges to complex textures. These learned features, represented as a dense feature map, are then flattened and embedded into a sequential patch representation suitable for transformer architecture input.

Once patch embeddings are generated, they are forwarded to the ViT module, which applies a series of transformer encoder blocks to model inter-patch dependencies and positional relationships. Through multi-head self-attention and feed-forward networks, ViT generates a high-dimensional latent representation that captures both the visual condition and aging indicators of the food item. The model is trained to classify each food image into one of four predefined freshness classes: Fresh, Fit for Consumption, About to Expire and Expired. This prediction assists in intelligent inventory decisions such as prioritizing consumption or removal, without the need for an explicit expiration timestamp. By leveraging both spatial precision and temporal reasoning, the hybrid VGG-19–ViT model shows strong potential in real-time food quality monitoring, dynamic freshness tracking and minimizing food waste in high-throughput environments like restaurants, canteens or retail kitchens.

E. Simulation Setup

The proposed hybrid VGG-19-ViT model was implemented in a high-performance computing environment configured with an Intel Core i7 processor, 32 GB RAM and an NVIDIA Tesla T4 GPU to ensure robust training and inference performance.

The model was developed using the TensorFlow framework with the Keras API, leveraging its modularity and advanced capabilities for implementing custom DL architectures. Model training, evaluation and prototyping were executed on Google Colaboratory Pro, which provided accelerated GPU support and cloud infrastructure optimizing both training time and memory efficiency. To ensure optimal learning, convergence speed and generalization capability, key hyperparameters were carefully selected through empirical tuning and preliminary examinations. Table II shows the hyperparameter specifications utilized in the proposed model.

TABLE II. HYPERPARAMETER SPECIFICATIONS

Hyperparameters	Values
Optimizer	ADAM
Activation Function	Softmax
Loss Function	Categorical Cross-entropy
Batch Size	32
Epochs	50
Learning Rate	0.0001
Dropout Rate	0.2
Number of Transformer Layers	12
Attention Heads (ViT)	12

IV. RESULTS

A set of standard evaluation metrics has been utilized for in depth performance evaluation of the proposed model, as shown in Eq. (18) to Eq. (22). True Positives (TP), False Positives (FP), True Negatives (TN) and False Negatives (FN) are mathematically computed using confusion matrix core elements. Different metrics offer unique insights of the model performance, overall correctness by accuracy, precision and recall highlight the ability of model to correctly predict food freshness stage without excessive misses or false alarms.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (18)$$

$$Arecision = \frac{TP}{TP+FP} \quad (19)$$

$$Recall = \frac{TP}{TP+FN} \quad (20)$$

$$F1\ score = 2 \times \frac{precision \times Recall}{Precision+Recall} \quad (21)$$

The % Food Waste Reduction calculates the percentage of food waste that is reduced as a result of the proposed model's timely forecasts, especially when products are identified as "About to Expire". It is calculated based on the weight of food consumed before expiration, via predictions by the proposed model, relative to the total food that would have been expired and wasted otherwise. The analysis considered standard portion sizes and average weight estimates for each food category, allowing a precise quantification of rescued food in kilograms. This indicator evaluates how the model helps food service settings manage waste and inventory in a sustainable way.

$$\%FWR = \frac{W_{baseline} - W_{model}}{W_{baseline}} \times 100 \quad (22)$$

where, $W_{baseline}$ and W_{model} are the total waste generated without and with the proposed model, respectively. A higher $\%FWR$ indicates greater effectiveness in reducing waste, showcasing the model's practical utility in inventory decision-making.

The accuracy plot of the proposed model, illustrated in Fig. 6, reveals a consistently improving learning trajectory over 50 epochs, underscoring the model's effective optimization. Initially, both training and validation accuracy rise sharply during the first 10 epochs, indicating rapid feature learning. The training accuracy steadily improves and reaches near-saturation beyond epoch 20, surpassing 99%, while the validation accuracy stabilizes just below 98%, suggesting robust generalization on unseen data. The gap between the two curves remains minimal, indicating low overfitting and well-regularized model behavior. This convergence confirms the proposed model's ability to learn meaningful representations across diverse food freshness categories and maintain high predictive reliability across both training and validation phases.



Fig. 6. Accuracy plot of the proposed model.

The loss plot of the proposed model, illustrated in Fig. 7, demonstrates the convergence behavior of the proposed model across 50 epochs. Both training and validation losses exhibit a sharp decline during the initial epochs, indicating effective learning and a rapid reduction in prediction error. The training loss continues to decrease steadily, approaching near-zero values by the final epochs, reflecting the model's high confidence on seen data. Meanwhile, the validation loss also follows a decreasing trend and stabilizes around 0.08, suggesting good generalization without signs of significant overfitting. The small and consistent gap between the two curves affirms the robustness and stability of the training process. Overall, the minimized loss values validate the model's capability to distinguish food freshness categories accurately and reliably.

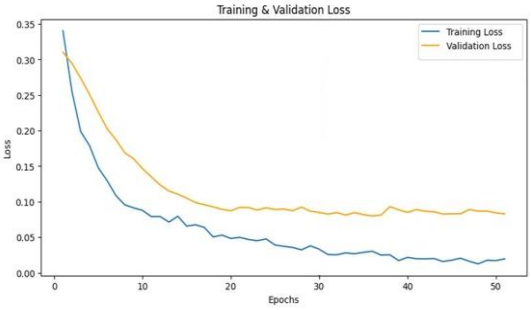


Fig. 7. Loss plot of the proposed model.

The classification report for the proposed model, illustrated in Table III, reflects strong overall performance across all food freshness categories. The model achieves excellent precision and recalls scores for the Fresh and Fit for Consumption classes (0.99/0.99 and 0.99/0.98, respectively), demonstrating excellent ability to consistently and accurately distinguish consumable items. The About to Expire class shows a precision of 0.97 and recall of 0.98, indicating reliable early identification of items nearing spoilage and the Expired class, often difficult to detect due to visual ambiguity, still attains solid performance with a precision of 0.94 and recall of 0.99, minimizing false negatives in disposal-critical cases.

TABLE III. CLASSIFICATION REPORT OF THE PROPOSED MODEL

	Precision	Recall	F1-score
Fresh	0.99	0.99	0.98
Fit for Consumption	0.99	0.98	0.98
About to expire	0.97	0.98	0.97
Expired	0.94	0.99	0.96
Accuracy	0.98		
macro avg	0.97	0.98	0.98
weighted avg	0.98	0.98	0.98

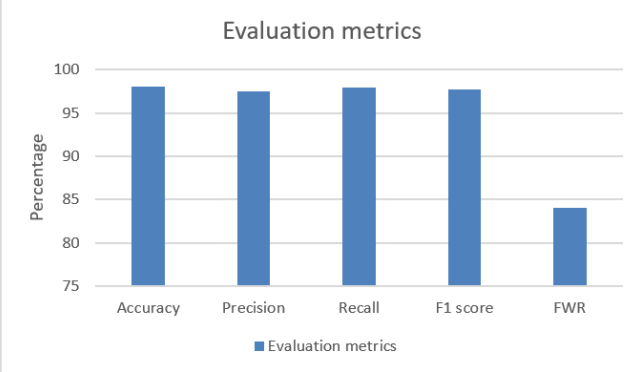


Fig. 8. Evaluation metrics of proposed model.

The evaluation metrics illustrated in Fig. 8 shows a 98% overall accuracy rate confirming that the model can accurately identify most food categories in the test dataset. With a precision score of 97.5%, the model minimizes false positives and proves to be quite dependable, while the 97.9% recall indicates how well the model captures almost all real, pertinent events, crucial for promptly identifying perishable or expired goods. Furthermore, the F1-score of 97.75% verifies a balanced performance, guaranteeing that predictions are accurate and comprehensive. The proposed model significantly enhanced food sustainability by enabling real-time freshness detection and shelf-life prediction, achieving a remarkable 84% reduction in food waste. The proposed model's real-time applicability in minimizing food loss across dynamic food service operations were demonstrated by its excellent capability to detect early signs of deterioration that allowed for timely consumption or redirection.

The confusion matrix illustrated in Fig. 9 analyses the performance of the proposed model across four food freshness classes. The model shows strong predictive accuracy, with particularly high true positive counts for Fresh (1443), Fit (for consumption) (1880), About to Expire (930) and Expired (448) categories. Misclassifications are minimal across all classes, with most errors occurring between neighboring freshness stages, reflecting the natural difficulty in distinguishing borderline cases. The overall accuracy of 98% confirms the model's robustness and high generalization capability in real-world food freshness classification.

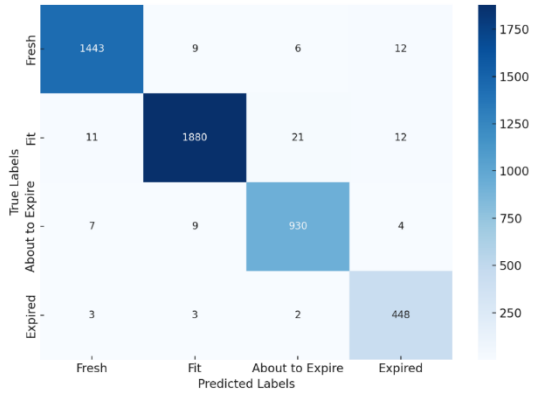


Fig. 9. Confusion matrix of the proposed model.

Fig. 10 represents the results of the proposed model, showcasing a variety of correctly predicted samples across all four freshness classes. Each image includes ground truth and predicted labels, demonstrating the model's ability to generalize across different food types and states with high visual accuracy.



Fig. 10. Results of the proposed model.

Table IV and Fig. 11 illustrate the comparative performance of different architectures applied to various food image datasets. InceptionResNetV2 trained on a Korean food image dataset achieved an accuracy of 81.91%, reflecting the challenges posed by diverse regional cuisine representations. ResNet-50 and Inception-V3, both evaluated on the Food-101 dataset, attained accuracies of 90%, but were constrained by dataset-level variability. EfficientNet V2L on the Indonesian food dataset recorded an accuracy of 85.44%, indicating limitations posed by complex food presentation styles. The SGLANet model applied to the ISIA Food-500 dataset demonstrated a validation accuracy of 90.92%. In contrast, the proposed VGG-19–ViT hybrid model applied to the Food Image Classification dataset achieved a substantially higher accuracy of 98%, clearly outperforming all other architectures. This result underscores the effectiveness of integrating convolutional feature extraction with transformer-based attention mechanisms and highlights the advantage of employing a freshness-annotated dataset tailored for food waste reduction.

TABLE IV. ACCURACY COMPARISON OF THE PROPOSED MODEL WITH EXISTING METHODS

Author [Ref]	Food Image Dataset Used	Method	Accuracy (%)
Chun et al. [10]	Korean Food Image Dataset	InceptionResNetV2	81.91
Louro et al. [11]	Food-101	ResNet-50	90
Rasyidi et al. [14]	Indonesian Food Dataset	EfficientNet V2L	85.44
Min et al. [19]	ISIA Food-500	SGLANet	90.92
Goh & Yann [22]	Food-101	Inception-V3	90
Proposed model	**Food Image Classification**	**VGG-19-ViT hybrid model**	**98**

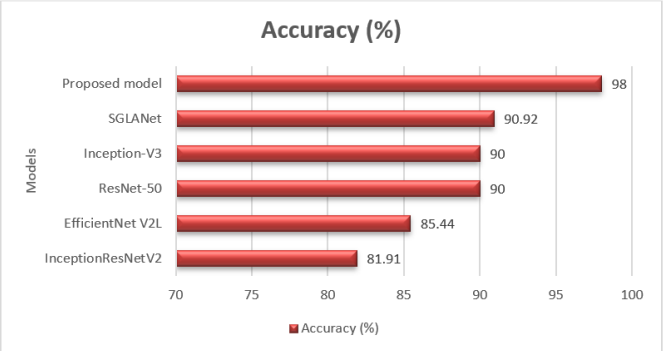


Fig. 11. Accuracy comparison of the proposed model with existing methods.

Table V and Fig. 12 illustrate the comparative performance of different approaches to food waste reduction, evaluated solely on the basis of the percentage of waste reduction achieved, irrespective of whether the methods relied on traditional survey-based strategies, machine learning forecasting, or real-time monitoring systems. Reported reductions vary widely, from 9.46% with SmartNoshWaste to intermediate values of 22% with neural networks and heuristic optimization, 30% through cloud-based machine learning with life cycle assessment, and 40% with AI-driven supply chain optimization. Higher reductions were observed in food service and institutional settings, with tracking systems in HORECA achieving 51% and forecasting models combining random forests and LSTM networks achieving 52%. Advanced predictive models showed further promise, with artificial neural networks reaching 79.66% and random forest–based planning attaining 82%. In comparison, the proposed VGG-19–ViT hybrid model delivered the highest performance with an 84% reduction in food waste, demonstrating the effectiveness of image-based freshness prediction in directly supporting sustainability goals.

TABLE V. % FWR COMPARISON OF THE PROPOSED MODEL WITH EXISTING METHODS

Author [Ref]	Model	% FWR
Dey et al. [12]	SmartNoshWaste	9.46%
Nascimento et al. [13]	RF	82%
Jacob et al. [15]	AI-Driven Optimization	40%
Hübner et al. [16]	Cloud ML + LCA	30%
Sigala et al. [17]	Waste Tracking in HORECA	51%
Wang [20]	NN, SVM with GA & PSO	22%
Rodrigues et al. [21]	RF, LSTM	52%
Faezirad et al. [23]	ANN	79.66%
Proposed model	**VGG-19-ViT hybrid model**	**84%**

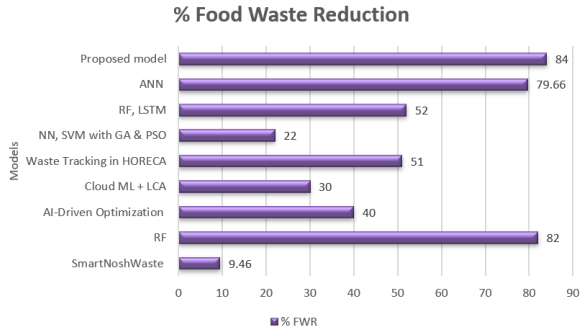


Fig. 12. % FWR comparison of the proposed model with existing methods.

V. DISCUSSION

The results clearly demonstrate that the proposed model delivers strong and reliable performance in food freshness classification. The very high precision and recall scores in the Fresh and Fit for Consumption classes confirm the model's ability to accurately identify items suitable for use, minimizing the risk of unnecessary disposal. The slightly lower precision observed in the Expired class, despite its excellent recall, indicates that the model tends to prioritize capturing all true expired items, which is advantageous for avoiding false negatives in disposal-critical cases. Similarly, the About to Expire class results reflect the model's strength in early spoilage detection, a key factor in reducing waste through timely redirection or consumption. The overall accuracy of 98% and the balanced F1-score highlight the stability of the model across categories. Beyond numerical accuracy, the observed 84% reduction in food waste illustrates the real-world impact of integrating such a system into food service operations, where early and dependable freshness prediction can transform management practices and contribute to sustainability goals.

VI. CONCLUSION

The study proposed a novel hybrid deep learning model integrating VGG-19 and Vision Transformer (ViT) architectures for intelligent monitoring of food freshness and predicting the remaining shelf-life of prepared food items. A meticulously selected food image dataset with 34 distinct food categories that represented actual restaurant and catering settings was employed in training the model. The framework successfully categorized food images into four freshness classes: Fresh, Fit for Consumption, About to Expire and Expired, by employing VGG-19 to extract deep visual characteristics and ViT to predict temporal degradation trends. In doing so, the study makes a theoretical contribution by demonstrating how convolution-based fine-grained feature extraction can be effectively fused with transformer-based self-attention to capture both spatial and temporal degradation cues, offering a new direction for freshness prediction models in food technology. The classification greatly reduced food waste and enabled prompt consumption decisions. The proposed model achieved 98% accuracy, 97.5% precision, 97.9% recall, 97.75% F1-score, and an estimated 84% reduction in food waste, underscoring its strong potential for real-world deployment. The method remains scalable and lightweight by integrating freshness estimation without requiring a large amount of sensor data. Nevertheless, the study is not without limitations. The reliance on a single image modality may restrict generalization under varied lighting or presentation conditions, and the dataset, while diverse, is smaller in scale compared to global benchmarks. Additionally, external factors such as storage temperature or humidity were not integrated into the model. Despite these constraints, the study contributes to the domain by establishing that purely image-based hybrid architectures can reliably predict freshness levels and directly translate to measurable sustainability outcomes in food service operations. Future research could explore integration with real-time kitchen inventory systems, multi-modal learning using sensors, and deployment in edge devices for low-resource settings. Further studies may also validate the framework in larger-scale, multi-institutional datasets and investigate explainability mechanisms to improve model transparency for end-users. Such extensions would further enhance the impact and global applicability of the proposed model in minimizing food waste.

REFERENCES

[1] M. Lai, A. Rangan, and A. Grech, “Enablers and barriers of harnessing food waste to address food insecurity: A scoping review,” Nutr. Rev., vol. 80, no. 8, pp. 1836–1855, Aug. 2022.

[2] A. Pandey, “Food wastage: Causes, impacts and solutions,” Sci. Herit. J., vol. 5, pp. 17–20, 2021.

[3] N. Aloysius, J. Ananda, A. Mitsis, and D. Pearson, “Why people are bad at leftover food management? A systematic literature review and a framework to analyze household leftover food waste generation behavior,” Appetite, vol. 186, p. 106577, Feb. 2023.

[4] L. Wei, G. Prabhakar, and L. N. Duong, “Usage of online food delivery in food waste generation in China during the crisis of COVID-19,” Int. J. Food Sci. Technol., vol. 58, no. 10, pp. 5602–5608, Oct. 2023.

[5] Y. Shigetomi, A. Ishigami, Y. Long, and A. Chapman, “Curbing household food waste and associated climate change impacts in an ageing society,” Nat. Commun., vol. 15, no. 1, p. 8806, Jan. 2024.

[6] C. Sundgren, “Circular supply chain relationships for food redistribution,” J. Clean. Prod., vol. 336, p. 130393, Jan. 2022.

[7] V. Amicarelli, A. C. Aluculesei, G. Lagioia, R. Pamfilie, and C. Bux, “How to manage and minimize food waste in the hotel industry: An exploratory research,” Int. J. Cult. Tour. Hosp. Res., vol. 16, no. 1, pp. 152–167, Jan. 2022.

[8] A. Režek Jambrak, M. Nutrizio, J. Dukić, I. Djekić, M. Vinceković, S. Jurić, and F. Donsi, “Digitalisation, bioinformatics, and delivery systems in sustainable nonthermal extraction of proteins,” Int. J. Food Sci. Technol., vol. 60, no. 1, p. vvae038, Jan. 2025.

[9] R. N. Arshad, Z. Abdul-Malek, C. Parra-López, A. Hassoun, M. I. Qureshi, A. Sultan, and G. Garcia-Garcia, “Food loss and waste reduction by using Industry 4.0 technologies: Examples of promising strategies,” Int. J. Food Sci. Technol., vol. 60, no. 1, p. vvaf034, Jan. 2025.

[10] M. Chun, H. Jeong, H. Lee, T. Yoo, and H. Jung, “Development of Korean food image classification model using public food image dataset and deep learning methods,” IEEE Access, vol. 10, pp. 128732–128741, 2022.

[11] J. Louro, F. Fidalgo, and Â. Oliveira, “Recognition of food ingredients—dataset analysis,” Appl. Sci., vol. 14, no. 13, p. 5448, Jul. 2024.

[12] S. Dey, S. Saha, A. K. Singh, and K. McDonald-Maier, “SmartNoshWaste: Using blockchain, machine learning, cloud computing and QR code to reduce food waste in decentralized web 3.0 enabled smart cities,” Smart Cities, vol. 5, no. 1, pp. 162–176, Mar. 2022.

[13] A. M. Nascimento, A. Queiroz, V. V. de Melo, and F. S. Meirelles, “Applying Artificial Intelligence to reduce food waste in small grocery stores,” Unpublished.

[14] M. A. Rasyidi, Y. S. Mardhiyyah, Z. Nasution, and C. H. Wijaya, “Performance comparison of state-of-the-art deep learning model architectures in Indonesian food image classification,” Bull. Electr. Eng. Inform., vol. 13, no. 5, pp. 3355–3368, Oct. 2024.

[15] L. A. Jacob and O. L. U. F. E. M. I. Omitoyin, “AI-driven farm-to-table supply chain optimization: A case study of reducing food waste,” J. Entomol. Agron. Stud., 2024.

[16] N. Hübner, J. Caspers, V. C. Coroamă, and M. Finkbeiner, “Machine-learning-based demand forecasting against food waste: Life cycle environmental impacts and benefits of a bakery case study,” J. Ind. Ecol., vol. 28, no. 5, pp. 1117–1131, May 2024.

[17] E. G. Sigala, P. Gerwin, C. Chroni, K. Abeliotis, C. Strotmann, and K. Lasaridi, “Reducing food waste in the HORECA sector using AI-based waste-tracking devices,” Waste Manag., vol. 198, pp. 77–86, Jan. 2025.

[18] T. Li and Y. Sun, “An intelligent food inventory monitoring system using machine learning and computer vision,” Unpublished, 2022.

[19] W. Min, L. Liu, Z. Wang, Z. Luo, X. Wei, X. Wei, and S. Jiang, “ISIA Food-500: A dataset for large-scale food recognition via stacked global-local attention network,” in Proc. 28th ACM Int. Conf. Multimedia, Oct. 2020, pp. 393–401.

[20] H. Wang, “Agricultural product supply chain inventory control and allocation decision support system integrating artificial intelligence technology,” Int. J. High Speed Electron. Syst., p. 2540556, 2025.

[21] M. Rodrigues, V. Migueis, S. Freitas, and T. Machado, “Machine learning models for short-term demand forecasting in food catering services: A solution to reduce food waste,” J. Clean. Prod., vol. 435, p. 140265, Jan. 2024.

[22] A. M. Goh and X. L. Yann, “Food-image classification using neural network model,” Int. J. Electron. Eng. Appl., vol. 9, no. 3, pp. 12–22, 2021.

[23] M. Faezirad, A. Pooya, Z. Naji-Azimi, and M. A. Haeri, “Preventing food waste in subsidy-based university dining systems: An artificial neural network-aided model under uncertainty,” Waste Manag. Res., vol. 39, no. 8, pp. 1027–1038, Aug. 2021.

[24] C. Malefors, L. Secondi, S. Marchetti, and M. Eriksson, “Food waste reduction and economic savings in times of crisis: The potential of machine learning methods to plan guest attendance in Swedish public catering during the COVID-19 pandemic,” Socio-Econ. Plan. Sci., vol. 82, p. 101041, Jan. 2022.

[25] https://www.kaggle.com/datasets/gauravduttakiit/food-image-classification

[26] T. H. Nguyen, T. N. Nguyen, and B. V. Ngo, "A VGG-19 model with transfer learning and image segmentation for classification of tomato leaf disease," AgriEngineering, vol. 4, no. 4, pp. 871–887, Dec. 2022.

[27] C. Xia, X. Wang, F. Lv, X. Hao, and Y. Shi, "ViT-COMER: Vision Transformer with convolutional multi-scale feature interaction for dense predictions," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2024, pp. 5493–5502.

Analysis of an RGB-D Simultaneous Localization and Mapping Algorithm for Unmanned Aerial Vehicle

Muhammad Zamir Fathi Mohammad Effendi[1], Norhidayah Mohamad Yatim[2], Zarina Mohd Noh[3], Nur Aqilah Othman[4]

G7 Aerospace Sdn. Bhd., Pusat Perindustrian Cipta Selatan, Bandar Serenia, 43900, Sepang, Selangor, Malaysia[1]
Fakulti Teknologi dan Kejuruteraan Elektronik dan Komputer (FTKEK), Universiti Teknikal Malaysia Melaka (UTeM), Durian Tunggal, Melaka, Malaysia[2, 3]
Faculty of Electrical & Electronics Engineering Technology, Universiti Malaysia Pahang Sultan Abdullah, Kuantan, Pahang, Malaysia[4]

***Abstract*—This study investigates the implementation of an RGB-D Simultaneous Localization and Mapping (SLAM) algorithm on an unmanned aerial vehicle (UAV) equipped with an Intel RealSense D435i camera. The study focuses on Real-Time Appearance-Based Mapping (RTAB-Map), a well-established RGB-D SLAM method capable of building 3D maps while simultaneously localizing a robot within its environment. Despite its advanced capabilities, deploying RTAB-Map on UAVs introduces specific challenges due to the dynamics of aerial navigation. This research evaluates the performance of RTAB-Map in terms of robustness, precision, and accuracy to optimize its application in UAV-based RGB-D SLAM. The findings reveal that the sequential frame-matching approach, combined with a minimum inliers threshold of 10, provides the most robust performance. In contrast, the global matching approach with a minimum inliers threshold of 20 offers better precision and accuracy. The results show that this implementation, utilizing an off-the-shelf hardware and software setup, has significant potential for advanced applications such as monitoring and surveillance in environments, where dense 3D mapping is critical.**

***Keywords*—Unmanned aerial vehicle; UAV; simultaneous localization and mapping; SLAM; RGB-D; real-time appearance based map; RTAB-map**

I. Introduction

Autonomous robotics has been a key area in the field of robotics. The necessity of determining a robot's position on a map in unknown environments, while simultaneously building the map representation itself, has led to the development of Simultaneous Localization and Mapping (SLAM) algorithms over the past three decades [1]-[4]. The demand for SLAM algorithms has grown significantly, as they are widely used for various applications in both ground-based and aerial robots. Computer vision technology has advanced over recent years, and the usage of vehicles with teleoperation capabilities is seen as a big potential to be commercialized. By integrating the SLAM algorithm with these vehicles, a technological milestone could be achieved. This study focused on the issue of SLAM with Unmanned Aerial Vehicle (UAV) which addresses a specific SLAM problem, involves the challenge of mapping the environment that has been explored by UAV in a 3D map representation, while determining its position within the gradually constructed 3D map [5].

Despite intensive research, implementing SLAM on UAV platforms remains challenging. One of the primary issues is that UAVs operate at higher speeds, which can reduce the accuracy of SLAM algorithms. Moreover, when applying SLAM in large or complex environments, UAVs may encounter difficulties such as rapid viewpoint changes, inconsistent loop closures, and accumulated drift errors [6]-[8]. A drawback often occurs when the UAV revisits previously mapped regions, where newly estimated landmark locations may not align with earlier ones. A robust and efficient algorithm is therefore required to achieve reliable SLAM performance for UAVs, especially given the trade-off between real-time operation and mapping accuracy. While robust methods exist for static and structured environments, mapping unstructured, dynamic, or large-scale settings continues to be an open research problem [2] [4].

Among existing RGB-D SLAM algorithms, RTAB-Map has proven effective as a graph-based approach in various robotic applications. However, its performance when applied to UAVs is inconsistent. UAV platforms introduce unique challenges such as high-speed motion, rapid viewpoint changes, and dynamic environments, which can significantly degrade SLAM accuracy and robustness. In particular, the performance of RTAB-Map depends heavily on parameter configurations (e.g., camera resolution, frame rate, odometry strategy, and loop closure thresholds) [9]-[12]. Yet, there is limited research systematically examining how these parameters influence UAV-based SLAM performance. Without such insights, it is difficult to identify optimal configurations that balance accuracy, robustness, and real-time feasibility for UAV operations.

Given this problem, this research study tries to address the following questions:

- How do state-of-the-art RGB-D SLAM algorithms such as RTAB-Map perform when applied to UAV platforms in terms of accuracy, robustness, and efficiency?
- Can modifications, parameter tuning, or integration strategies improve their performance for UAV-based applications?

To address the research questions outline the following objectives are outlined:

- To evaluate the impact of key parameters such as RGB and depth camera resolution, frame rate (fps), visual odometry strategies, and loop closure inlier thresholds on the performance of RTAB-Map for UAV-based SLAM.
- To investigate the optimum parameter configuration that maximizes accuracy, efficiency, and robustness of RTAB-Map in UAV navigation and mapping tasks.

This research is significant as it reduces the gap between conventional SLAM implementations and the unique requirements of UAV navigation. By evaluating and comparing RGB-D SLAM algorithms in UAV contexts, this study is expected to provide researchers and practitioners with some insights into algorithmic strengths and weaknesses, as well as practical guidance for deploying UAV-based SLAM in real-world environments. It is hoped that the outcomes can support the development of more reliable and efficient autonomous UAV systems for applications in surveillance, mapping, disaster response, and environmental monitoring.

The remainder of this study is structured as follows: Section II presents the literature review of the SLAM algorithm, focusing on two prominent RGB-D SLAM algorithms: ORB-SLAM2 and RTAB-Map. Section III outlines the research methodology applied in this study, while Section IV discusses the results and findings of the experiments conducted. Finally, Section V provides the conclusion of the study.

II. Literature Review

A. Visual Simultaneous Localization and Mapping (VSLAM)

In the Simultaneous Localization and Mapping (SLAM) algorithm, mapping is for obtaining a model of the robot environment, and localization is to estimate the position of the robot in the obtained map. Despite significant progress in this area, it still poses great challenges [2]-[4]. The accuracy of environment mapping depends highly on the effective parameters which need to be studied. The important parameters that affect SLAM performance include sensor uncertainty, correspondence issue, loop closing, time complexity and dynamic environment [1].

Visual sensor implementation of SLAM or also known as visual SLAM (VSLAM) systems, can be classified into monocular SLAM, binocular SLAM and RGB-D SLAM [9]. The input sensor for classic visual SLAM systems is a monocular or stereo camera. Using these two vision sensors to reconstruct a 3D map, sophisticated map initialization and map point triangulation operations are required, which are computationally challenging and can result in noisy observations. The introduction of RGB-D cameras, which deliver RGB images along with depth information, addresses the issue that comes with monocular and stereo cameras. Due to the fact that RGB-D cameras can provide both colored and depth images simultaneously, they are increasingly utilized for indoor scene reconstruction and can effectively address limitations such as reconstruction of low-textured areas [12]. Low-textured areas are surfaces with few distinguishing features. These areas are difficult to reconstruct because they lack sufficient feature points for effective tracking and mapping [10], [11].

B. ORB-SLAM2 and RTAB-Map

In this study, the RGB-D SLAM algorithms of interest that are to be researched are RTAB-Map and ORB-SLAM2. RTAB-Map was first developed in 2009 and released as an open source library in 2013. Since then, it has evolved into a complete graph-based SLAM approach that can be used in a variety of setups and applications [13]. RTAB-Map is a graph-based SLAM approach based on an incremental appearance-based loop closure detector. To determine whether a new image is likely to come from a prior or new place, the loop closure detector employs a bag-of-words approach. When a loop closure hypothesis is accepted, a new constraint is added to the map's graph, and the map's mistakes are minimised using a graph optimizer. To ensure that real-time limitations on large-scale environments are always maintained, a memory management strategy is employed to limit the number of locations used for loop closure detection and graph optimization. Visual odometry strategies play a crucial role in RTAB-Map's operation. Frame-to-frame odometry compares each image to the previous one, offering speed and efficiency but suffering from drift over time. In contrast, frame-to-map odometry compares each image to a gradually built map, improving accuracy but increasing computational demands [11], [14].

ORB-SLAM2 is a comprehensive SLAM system with map reuse, loop closing, and re-localization capabilities. Proposed in 2016 by Raul Mur-Artal et al., ORB-SLAM2 provides support for stereo cameras and RGB-D cameras as an improvement from its predecessor, ORB-SLAM. The algorithm is able to operate swiftly in real-time [9]. ORB-SLAM2 selects ORB as the 2D features for all SLAM operations, including tracking, mapping, relocalization, and loop closing, in order to maintain speed and rotation invariance [10]. If tracking was successful in the previous frame, ORB-SLAM2 use a constant velocity motion model to project the camera pose and conduct a guided search of the map points encountered in the previous frame. If inadequate matches were discovered (i.e., due to an inadequate motion model), ORB-SLAM2 does a broader search of the map points surrounding the subject's position in the previous frame. The pose is then enhanced using bundle adjustment (BA) through the discovered 2D correspondences [10].

Although ORB-SLAM2 has significantly improved in both efficiency and precision, certain issues remain unresolved. One major challenge is system tracking failure in dynamic environments, which arises from the feature point extraction process being insufficiently robust in conditions such as sudden lighting changes, excessively strong or weak light intensity, or environments with minimal texture information [9]. Solving motion blur becomes a key focus in visual SLAM since common cameras frequently produce image blur when moving rapidly.

N. Ragot et al. (2019) conducted research on the benchmark of the two reviewed SLAM algorithms; ORB-SLAM2 and RTAB-Map [15]. The research utilized an Intel RealSense D435 camera mounted on a robotics-powered electric wheelchair to test and evaluate the performance of these algorithms across various configurations, such as straight-line motion, straight-line motion with a return path, and circular paths with loop closure. All experiments were performed in a controlled indoor environment.

The study highlighted that both RTAB-Map and ORB-SLAM2 have unique strengths in specific aspects. RTAB-Map, a graph-based SLAM system, integrates two key algorithms: loop closure detection and graph optimization. According to N. Ragot et al., RTAB-Map demonstrated more accurate trajectory estimation, better performance in translational movements, and more effective utilization of RGB data. However, it produced less accurate results during rotational movements. On the other hand, ORB-SLAM2 relies on consecutive image frames to construct a map and localize itself within it. It showed superior performance in distance measurement and odometry estimation compared to RTAB-Map. Nonetheless, its performance with RGB cameras was less effective than stereo cameras, particularly in outdoor scenarios [9].

While these benchmarking studies provide valuable insights, most evaluations of RTAB-Map have concentrated on ground robots or controlled indoor environments with fixed parameter settings. Consequently, there remains a limited understanding of how RTAB-Map behaves in UAV contexts, where challenges such as higher flight speeds, rapid viewpoint changes, and real-time processing demands are more critical. This gap shows the need for investigations into how key parameters such as camera resolution, frame rate (FPS), odometry strategy, and loop closure thresholds affect RTAB-Map's performance when deployed on UAVs. Addressing this gap is important for adapting RTAB-Map to aerial robotics and ensuring robust autonomous navigation in real-world scenarios.

Hence, this study focuses exclusively on RTAB-Map and conducts a parameter sensitivity analysis to investigate the optimum configuration for UAV-based SLAM.

III. Methodology

A. System Interface

In this study, a UAV modelled NXP KIT-HGDRONEK66 was used, where the devices for communication and flight control using PX4 [16] flight management unit were on board. Fig. 1 shows the front and top views of the assembled UAV. All devices are mounted on the UAV bar, and the camera is attached underneath the UAV at its centre of gravity to ensure the UAV's stability. The overall system interface is illustrated in Fig. 2.

On board the UAV, a Raspberry Pi 4B was mounted to transmit data from the Intel RealSense D435i RGB-D camera [17] to the remote PC. Both the Raspberry Pi 4 and the remote PC run Ubuntu and Robot Operating System (ROS) as the system employs a ROS-based communication architecture. The remote PC is configured as the ROS master. The remote PC also runs SLAM using RTAB-Map. The Raspberry Pi on the UAV operates as a ROS node, publishing data from the RealSense camera to the RTAB-Map node on the remote PC.

In addition to data transmission, telemetry communication between the UAV and the remote PC is established using the Holybro SiK Telemetry Radio V3 device. This communication link enables real-time monitoring of flight parameters and the ability to issue control commands through the QGroundControl software on the remote PC.



Fig. 1. Front view (left) and top view (right) of the assembled UAV.

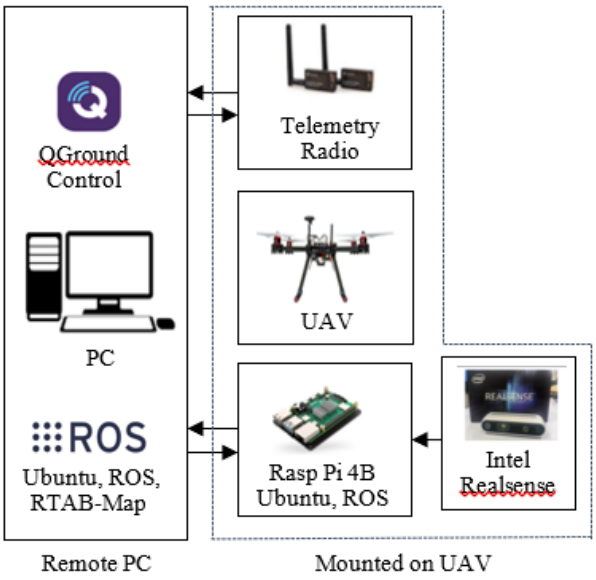


Fig. 2. Overall system interface of the RGB-D SLAM system.

B. Effective Parameters Tuning

The goal of this parameter study is to identify the optimum configuration of parameters that yield better performance of the RTAB-Map SLAM algorithm for UAVs. The parameters that were studied and tuned are as follows:

- RGB and depth camera resolution,
- RGB and depth camera number of fps,
- visual odometry strategy, and
- The minimum inliers threshold for loop closure detection

The resolution of RGB and depth cameras are parameters that study the correlation between the quality of images with transmission rate performance. A higher camera resolution yields a sharper image and gives more visual features in the surroundings. However, a higher quality image requires more processing and hence has an impact on the transmission rate or speed of the data sent to the ROS topics.

As for the RGB and depth camera fps, they are important parameters as they affect the frequency of data being published to the ROS topics. If there are more frames per second captured by both RGB and depth cameras, the data needs to be published to the RGB and depth topics more often. Hence, requires a higher transmission frequency.

In RTAB-Map, the RGB and depth topics are used to estimate the trajectory of the RGB-D sensor and to build a 3D map. Fig. 3 shows a redrawn rqt graph from ROS, considering only the relevant nodes and topics involved in this process. For better distinction, only the nodes are colored. The rqt graph shows that the /camera/realsense2_camera_manager node publishes all sensor data, including RGB, depth, and IMU information. This data is then processed by other nodes. The /rtabmap/rgbd_odometry node subscribes to the color and depth topics to perform visual odometry, estimating the localization of the RealSense camera. This localization is considered the UAV's position and is published to the /rtabmap/odom topic. The /rtabmap/rtabmap node subscribes to /rtabmap/odom, color, depth, and IMU data to build a 3D map and localize the UAV within the global map. It should be noted that the /rtabmap/rgbd_odometry node computes local odometry incrementally, the /rtabmap/rtabmap node performs global pose graph optimization and also loop closure detection. The mapping results, including the optimized trajectory and 3D map, are published on the /rtabmap/mapData topic.

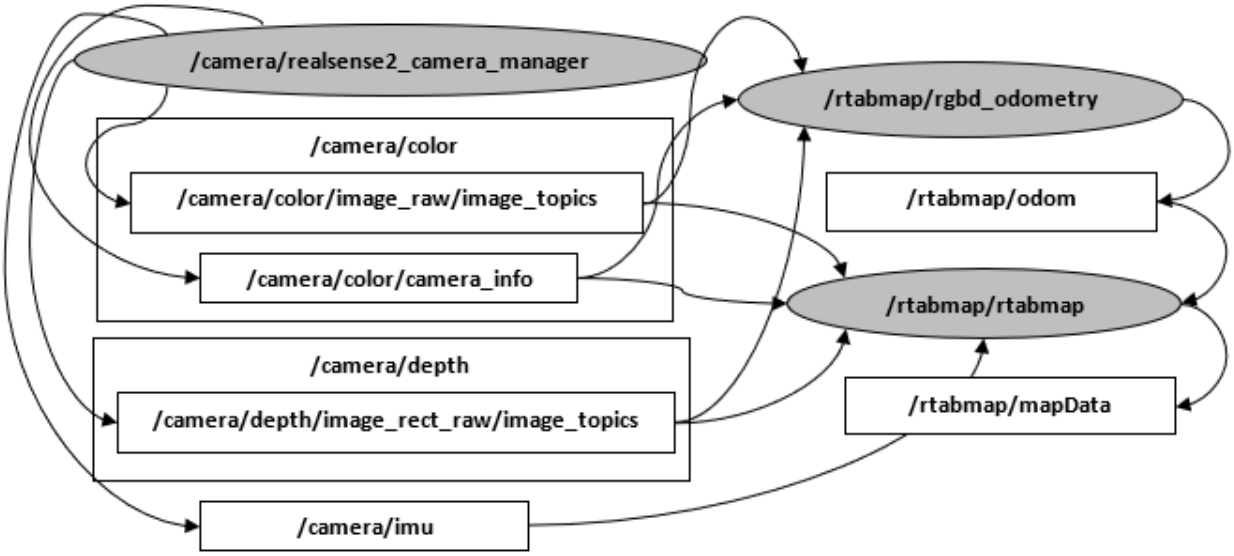


Fig. 3. A part of rqt graph when executing RTAB-Map.

The trajectory of the UAV is estimated using the visual odometry strategy implemented [14]. Visual odometry is the process of determining odometry information using sequential camera images to estimate the distance travelled by the RGB-D sensor on board, hence the UAV. The accuracy of visual odometry affects the overall performance of RTAB-Map. In this study, the visual odometry strategy was studied to determine its significance of impact. Visual odometry strategy that were tested includes frame-to-frame and frame-to-map.

In frame-to-frame odometry, the algorithm compares each image frame (or camera view) directly to the previous frame to estimate the trajectory. This is computed in the /rtabmap/rgbd_odometry node. Although this approach is efficient, as only a few consecutive images are compared, however, it is more prone to error accumulation over time. This strategy is suitable for real-time applications where some drift is acceptable. In frame-to-map odometry, the algorithm compares the current frame to an existing global map, which has been built gradually from past frames. This is computed in /rtabmap/rtabmap node. It is a more accurate approach as it references the entire map, but it can be slower and more computationally intensive. This approach is suitable for long-term mapping.

The last parameter studied is, minimum inliers threshold. It encapsulates how well the algorithm accepts input data as loop closure. Theoretically, a higher value allows the algorithm to only accept frames with more features to perform loop closure. This will result in lesser loop closures being detected as more matching visual features are required.

C. Performance Analysis

This section highlights the corresponding metrics that were observed and measured for performance evaluation purposes. The performance of the RGB-D SLAM algorithm was evaluated through a series of different categories, which are computational efficiency, robustness, precision and accuracy. Computational efficiency measures the amount of time required to process and transmit data. Metrics that were observed for this criterion are the publishing time of topics and the rate of data transmission. The ability of the algorithm to handle noise, outliers and other sources of errors defines the robustness criterion, where it is represented by the percentage of successful runs. A successful run is defined by a situation where a loop closure is successfully detected during a mapping sequence, and there are no odometry loss occurrences. The precision criterion measures the resolution and detail of the map produced by the algorithm. Keypoint detection per millisecond is observed to determine the precision of the algorithm. Last but not least, total distance travelled and distance between landmarks are the metrics measured to determine the accuracy of the algorithm, where the distance computed by the algorithm is compared with the ground truth.

IV. Results and Discussion

This section encapsulates the results obtained from experiments conducted based on the methodology mentioned in

previous section. The results were split based on different environments to observe the performance of the algorithm. Analysis on the findings of the experiments were discussed in this section. The effects of parameter tuning can be demonstrated through the analysis conducted.

A. Camera Parameter Optimization

Through research and experiments, it is noted that the publishing rate of topics is dependent on the Wi-fi signal, resolution and fps of the RGB and depth camera. The initial configuration is shown in the middle column of Table I. In the initial configuration, no data was published to the camera topics, preventing the algorithm from running. After further investigation, these parameters need to be tuned in order to yield better performance in terms of topics' publishing rate. The Wi-fi signal band of 2.4 GHz is better for long-range applications but has a slower speed, while 5 GHz gives significant improvement in speed but is lacking in terms of range coverage.

The fps of the camera, on the other hand, affects the transmission performance. This is because higher fps results in the camera capturing more frames that need to be processed every second. This may lead to a bottleneck, where the rate of frames being processed to be published as ROS topics exceeds the system's computational capacity, causing frames to be dropped and not transmitted. Overloading of the processing performance causes frames to be dropped and not transmitted. After switching to the 5 GHz band and reducing the fps of both the RGB and depth camera to 15 fps, the publishing rate of the topics was improved to an average of 7 Hz.

Further tuning of the parameter was conducted to obtain the best publishing rate performance. By using a lower setting of resolution for both RGB and depth camera, an experiment was conducted to observe the publishing rate. The new configuration that was tested can be observed in the rightmost column of Table I. As a result, the publishing rate of both the RGB and depth topics was further improved to around 15 Hz, which could be observed in Fig. 4 and Fig. 5. It can be concluded that transmission performance is directly affected by the resolution of RGB and depth cameras. This shows that the RGB and depth topics require minimal delay or bottleneck when processing the topic to be able to publish at an acceptable and consistent rate.

B. Indoor Environment Mapping Experiment

The RTAB-Map algorithm was first run on a Raspberry Pi without integration to the UAV. The Intel RealSense D435i camera was connected to the Raspberry Pi 4B, and handheld mapping was conducted inside our research laboratory in the faculty. This experiment was conducted to analyze the effect of parameter tuning in an indoor environment in terms of precision and accuracy, as well as to identify the most robust parameter configuration. The experiment was successfully conducted, where a 3D map and an occupancy grid map of the laboratory were obtained. Occupancy grid map is a map representation that is commonly used by the SLAM algorithm that generate precise metric maps which are close to the detailed environmental representations [7]. Fig. 6 and Fig. 7 show both maps, respectively.

TABLE I. INITIAL AND UPDATED CONFIGURATION

Parameter	Initial Value	Updated Value
Wi-fi band	2.4 GHz	5 GHz
RGB camera fps	30	15
Depth camera fps	30	15
Resolution of RGB camera	640 x 480	424 x 240
Resolution of depth camera	640 x 480	480 x 270

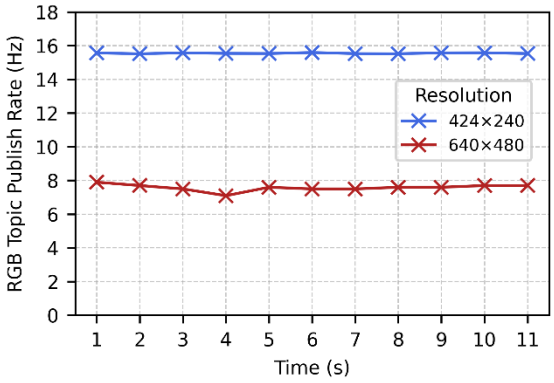


Fig. 4. Graph of publish rate of RGB topic vs. time.

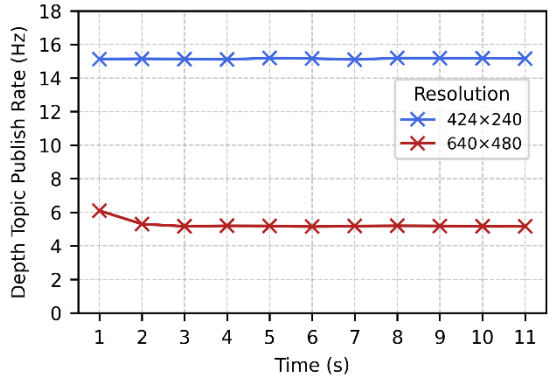


Fig. 5. Graph of publish rate of depth topic vs. time.

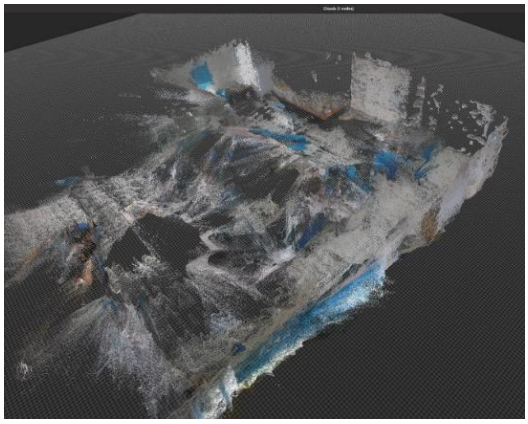


Fig. 6. 3D map of handheld mapping.

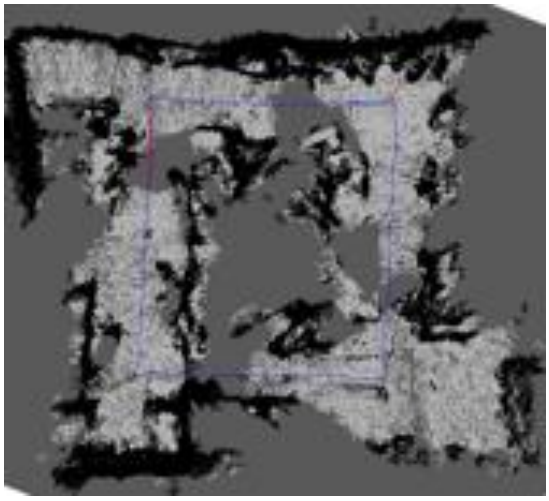


Fig. 7. Occupancy grid map of handheld mapping.

The experiment was conducted through a series of different parameter configurations. The parameters that are tuned were the visual odometry strategy and the minimum inliers threshold. As mentioned in Section III B, there are two visual odometry strategies, which are frame-to-map and frame-to-frame. On the other hand, the values used for the minimum inliers threshold were 10 and 20. Firstly, the robustness of the algorithm with these different parameter configurations was observed. In a total of 11 trials for each parameter configuration, the percentage of successful runs was computed. It was identified that the frame-to-frame odometry strategy with a minimum inliers threshold value of 10 showed the best performance in terms of robustness of the algorithm, with a 90.91% rate of successful runs.

We can observe that by changing the visual odometry strategy from frame-to-map to frame-to-frame, for each minimum inliers threshold, the value increases the robustness performance by at least 9.09%, as depicted in the bar chart in Fig. 8. The advantage of using a frame-to-frame setting over a frame-to-map is that the tracked features don't dip below a ratio of the previous frame the robot had seen. Hence, if the robot is not moving, the key frame would remain static and would remember the features in that frame. In this indoor environment experiment, the frame-to-frame odometry strategy provided better odometry compared to the frame-to-map odometry strategy, which indirectly increases the loop closure detection of the algorithm.

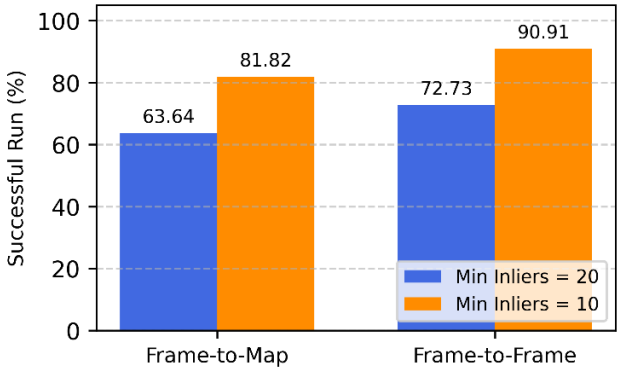


Fig. 8. Percentage of successful runs in the indoor environment.

Next, we can also observe from the bar chart in Fig. 8 that, by lowering the minimum inliers threshold from 20 to 10 for each visual odometry strategy boosts the robustness by at least 18.18%. A lower minimum inliers threshold results in frames with less features to be accepted as keyframe, which are then used for computation of localization and mapping by RTAB-Map algorithm. When the keyframe acceptance is more lenient, then the running performance of the RTAB-Map algorithm could be increased, as shown in the indoor environment mapping experiment.

The experiment was also conducted to evaluate the performance in terms of precision. For the frame-to-map visual odometry strategy, the keypoint detection per ms was observed and tabulated in the first two rows of Table II. The subsequent rows show the keypoint detection per ms for the frame-to-frame visual odometry strategy. For both strategies, a minimum inliers threshold value of 20 shows a higher keypoint detection per ms value compared to using a threshold value of 10. The observed results show that when the minimum inliers threshold is set to a higher value, the keypoint detection per ms will be higher. This is because a higher number of features need to be recognized within a frame for it to be accepted as a keyframe before proceeding to evaluate the next frame. Consequently, these selected frames can be considered of higher quality as they show to have more keypoints (refer to Table II). It can be concluded that applying a stricter threshold ensures the extraction of more keypoints, resulting in a more precise computation of the surrounding environment.

TABLE II. KEYPOINT DETECTION PER MS

Visual odometry	Min inliers threshold	Keypoint/ms
Frame-to-map	20	5.83
	10	5.20
Frame-to-frame	20	5.80
	10	4.86

Evaluation of performance in terms of accuracy was also conducted, where total distance travelled and the distance between two landmarks: Landmark 1 and Landmark 2, in meters, were observed and compared relative to the ground truth. Table III shows the estimated distance of the RTAB-Map algorithm. This shows that using RTAB-Map, similar to other common SLAM algorithms, the error is accumulated over the distance travelled. In this experiment, the error in the estimated trajectory is approximately 9.1 meters, which corresponds to about 26.6% error relative to the actual distance of 34.23 meters. In contrast, for a single measurement between Landmark 1 and 2, the error is only 0.15 meters, or approximately 7.5%, indicating that initial measurements are typically more accurate than long trajectory estimation.

TABLE III. DISTANCE ERROR FOR INDOOR ENVIRONMENT

Experiment	Actual (m)	RTAB-Map (m)	Error
Distance travelled	34.23	43.33	0.266
Landmark 1 and Landmark 2	2.00	2.15	0.075

Overall, it can be concluded that parameter configuration with the frame-to-frame visual odometry strategy and a minimum inliers threshold of 10 yields the most robust performance. This is due to the fact that it has higher trajectory accuracy and more lenient keyframe acceptance. However, there is a trade-off in terms of precision and accuracy where the generated 3D map is less dense and has fewer keypoints compared to the frame-to-map odometry strategy with a minimum inliers threshold of 20 parameter configuration.

C. Outdoor Environment Mapping Experiment

Before outdoor environment mapping is conducted using a UAV, an experiment with handheld mapping was first conducted to analyze the performance of the algorithm with a different layout. The purpose of this is to get the basis of the performance of the RTAB-Map algorithm in an outdoor environment. Since the nature of UAVs' flight is high-speed movement and maneuverable, a robust algorithm is desired to ensure a sustainable mapping process. Hence, the parameter configuration with the highest success rate was chosen, which was the frame-to-frame and minimum inliers threshold of 10. The experiments were executed at two different outdoor scenarios around the faculty: outdoor scenario 1 and outdoor scenario 2. As a result of the experiment, the 3D maps were successfully obtained for both of the outdoor scenarios, as depicted in Fig. 9 and Fig. 10.

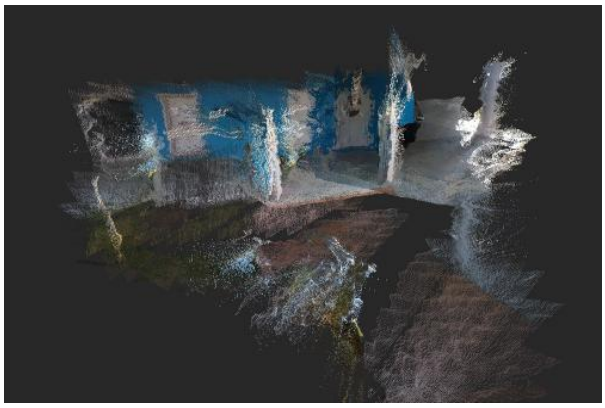


Fig. 9. 3D Map for outdoor 1.

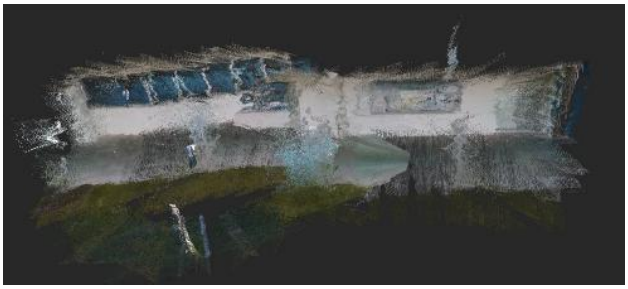


Fig. 10. 3D Map for outdoor 2.

Scenario of outdoor 1 and outdoor 2 yields different performance in terms of robustness and accuracy. The mapping sequence showed a 50% success rate for outdoor 1, while outdoor 2 demonstrates an 83.33% success rate out of a total of 6 trial runs, as shown in Fig. 11. In terms of accuracy, the total distance travelled, in meters, was measured and compared with the ground truth, as shown in Table IV. Outdoor 1 showcase a higher error (0.8283) compared with outdoor 2 (0.5676). Both these errors are still high and show that the accuracy of the trajectory travelled using RTAB-Map is less accurate. The longer the total distance travelled, increases the inaccuracy of the generated map due to accumulated error.

However, the computed error between landmarks remains relatively low for both outdoor 1 and outdoor 2, with values of 0.0259 and 0.0268, respectively (see Table V). This demonstrates that RTAB-Map is more reliable for short-term mapping that does not involve long distances travelled, as it can tolerate small drifts and still maintain a consistent map.

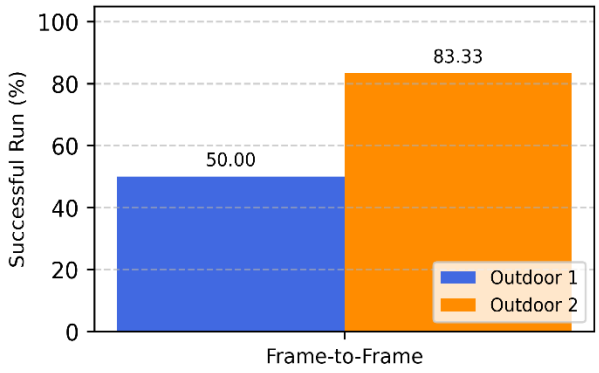


Fig. 11. Percentage of successful runs in outdoor environment.

TABLE IV. TRAVELLED DISTANCE ERROR FOR OUTDOOR ENVIRONMENT

Experiment	Actual (m)	RTAB-Map (m)	Error
Outdoor 1	16.53	30.22	0.8283
Outdoor 2	14.3	22.42	0.5676

TABLE V. DISTANCE BETWEEN LANDMARK ERROR FOR OUTDOOR ENVIRONMENT

Experiment	Actual (m)	RTAB-Map (m)	Error
Outdoor 1	3.18	3.26	0.0259
Outdoor 2	3.67	3.76	0.0268

V. LIMITATION

Although this study provides some insights into the effect of parameter configurations on RTAB-Map performance for UAV-based SLAM, several limitations remain. First, the experiments were conducted under controlled conditions with specific UAV hardware such as Intel RealSense D435i camera. The results may therefore not fully generalize to other UAV platforms, sensors, or outdoor environments subject to more dynamic conditions such as wind and lighting variations. Second, the evaluation was limited to a selected set of parameters; camera resolution, frame rate, visual odometry strategy, and loop closure inliers. While these parameters are critical, RTAB-Map contains additional configuration options (e.g., memory management strategies, feature extraction settings, and graph optimization techniques) that could further influence performance. Third, computational resources were constrained by the onboard UAV hardware, which may have affected the algorithm's scalability in larger or more complex environments.

VI. Conclusion and Future Works

RTAB-Map is an RGB-D SLAM algorithm with a loop closure detector that employs a bag-of-words approach to determine the possibility of a fresh image to be coming from a prior or new place. In this research, the loop closure detector of RTAB-Map utilizes both RGB and depth data from Intel RealSense D435i in making the decision whether to accept or deny loop closure hypotheses.

This project generally studies the applicability and relevance of SLAM algorithm implementation with UAVs by collecting all the important data and conducting a thorough assessment. Effective parameters and their relative effects on the performance of the SLAM algorithm were evaluated accordingly. The effective parameters were identified as RGB and depth camera resolution, RGB and depth camera fps, visual odometry strategy and minimum inliers threshold. The effect of tuning the parameters was then investigated in different environments (indoor and outdoor) to better understand the significance of the impact. An analysis was conducted to determine the best configuration of parameters to be applied for outdoor environment mapping by the UAV. Frame-to-frame odometry strategy minimum inliers threshold value of 10, proved to be the most robust setting to be implemented for UAV. However, there will be repercussions faced by the algorithm, where the performance will suffer in terms of precision and accuracy. Although the optimal parameters were identified for maximum robust performance of RTAB-Map, it still was not enough to cope with the movement of the UAV, as sharp turns would cause the algorithm to halt due to odometry data being lost.

Future research can extend this work in several directions. A broader exploration of RTAB-Map parameters and their interdependencies would provide deeper insights into optimizing UAV-based SLAM. Furthermore, integrating adaptive parameter tuning methods, such as machine learning-based auto-configuration, could allow UAVs to dynamically adjust SLAM parameters in response to environmental changes. Finally, long-term autonomous flight experiments, including large-scale outdoor mapping and multi-UAV collaborative SLAM, represent promising directions for advancing the robustness and applicability of RTAB-Map in real-world UAV operations.

Acknowledgment

The authors would like to thank the Centre for Research and Innovation Management (CRIM), Centre for Telecommunication Research & Innovation (CeTRI), Fakulti Teknologi dan Kejuruteraan Elektronik dan Komputer (FTKEK), Universiti Teknikal Malaysia Melaka (UTeM) and the Ministry of Higher Education Malaysia for the grant FRGS/1/2023/ICT02/UTEM02/2 which has made this research possible.

References

[1] M. Mahrami, M. N. Islam, and R. Karimi, “Simultaneous Localization and Mapping: Issues and Approaches,” 2013.

[2] S. Thrun and J. J. Leonard, “Simultaneous Localization and Mapping.”

[3] S. Wang, Z. Wu, and W. Zhang, “An overview of SLAM,” in Proc. 2018 Chinese Intelligent Systems Conference: Volume I, 2018, pp. 673–681.

[4] R. Azzam, T. Taha, S. Huang, and Y. Zweiri, “Feature-based visual simultaneous localization and mapping: a survey,” SN Applied Sciences, vol. 2, no. 2, pp. 1–20, Feb. 2020, doi: 10.1007/s42452-020-2001-3.

[5] T. T. O. Takleh, N. A. Bakar, S. A. Rahman, R. Hamzah, and Z. A. Aziz, “A brief survey on SLAM methods in autonomous vehicles,” Int. J. Eng. Technol. (UAE), vol. 7, no. 4, pp. 38–43, 2018, doi: 10.14419/ijet.v7i4.27.22477.

[6] Huijing Z., Masaki C., Ryosuke S., Xiaowei S., Jinshi C., and Hongbin Z., “SLAM in a Dynamic Large Outdoor Environment using a Laser Scanner,” IEEE Xplore, 2008.

[7] S. Saat, W. N. Abd Rashid, M. Z. M. Tumari, and M. S. Saealal, “Hectorslam 2D mapping for simultaneous localization and mapping (SLAM),” in Journal of Physics: Conf. Series, vol. 1529, no. 4, p. 042032, 2020, doi: 10.1088/1742-6596/1529/4/042032.

[8] M. Essaaidi, M. Nemiche, and IEEE Morocco Section, “SLAM algorithms implementation in a UAV, based on a heterogeneous system: A survey.”

[9] K. Luo, M. Lin, P. Wang, S. Zhou, D. Yin, and H. Zhang, “Improved ORB-SLAM2 Algorithm Based on Information Entropy and Image Sharpening Adjustment,” Math Probl Eng, vol. 2020, 2020, doi: 10.1155/2020/4724310.

[10] S. Zhang, L. Zheng, and W. Tao, “Survey and Evaluation of RGB-D SLAM,” IEEE Access, vol. 9, pp. 21367–21387, 2021, doi: 10.1109/ACCESS.2021.3053188.

[11] Z. Chen, W. Sheng, G. Yang, Z. Su, and B. Liang, “Comparison and analysis of feature method and direct method in visual SLAM technology for social robots,” in Proc. 2018 13th World Congress on Intelligent Control and Automation (WCICA), Jul. 2018, pp. 413–417, doi: 10.1109/WCICA.2018.8630714.

[12] B. Bacca, J. Salvi, and X. Cufí, “Appearance-based SLAM for mobile robots,” Artificial Intelligence Research and Development, pp. 55–64, 2009.

[13] M. Labbé and F. Michaud, “RTAB-Map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation,” J Field Robot, vol. 36, no. 2, pp. 416–446, Mar. 2019, doi: 10.1002/rob.21831.

[14] C. Audras, A. I. Comport, M. Meilland, and P. Rives, “Real-time dense RGB-D localisation and mapping,” in Proc. Australian Conf. on Robotics and Automation (ACRA), 2011.

[15] N. Ragot, R. Khemmar, A. Pokala, R. Rossi, and J.-Y. Ertaud, “Benchmark of visual SLAM algorithms: ORB-SLAM2 vs RTAB-Map,” in Proc. 2019 Eighth Int. Conf. on Emerging Security Technologies (EST), pp. 1–6, 2019.

[16] PX4, “PX4 robotic drone vehicle / flight management unit (FMU),” 2022. [Online]. Available: https://px4.io.

[17] Intel Corporation, “Intel® RealSense™ Camera D400 Series Product Family Datasheet Rev. 01/2019,” 2019. [Online]. Available: https://www.intel.com/content/dam/support/us/en/documents/emerging-technologies/intel-realsense-technology/Intel-RealSense-D400-SeriesDatasheet.pdf.

In-Depth Comparison of Supervised Classification Models - Performance and Adaptability to Practical Requirements

Mouataz IDRISSI KHALDI[1], Allae ERRAISSI[2], Mustapha HAIN[3], Mouad BANANE[4]
Laboratory of Artificial Intelligence & Complex Systems Engineering,
Hassan II University, ENSAM, Casablanca, Morocco[1, 3, 4]
Chouaib Doukkali University, El Jadida, Morocco[2]

***Abstract*—In this paper, we carried out an in-depth comparative analysis of five major supervised classification algorithms: Naïve Bayes, Decision Tree, Random Forest, KNN and SVM. These models were evaluated through a rigorous literature review, based on 20 criteria grouped into five key dimensions: algorithm performance, computational efficiency, practicality and ease of use, data compatibility and practical applicability. The results show that each algorithm has specific strengths and limitations: SVM and Random Forest stand out for their robustness and accuracy in complex environments, while Naïve Bayes and Decision Tree are appreciated for their speed, simplicity and interpretability. KNN, despite its intuitive approach, suffers from high complexity in the prediction phase, limiting its effectiveness on large datasets. This study aims to provide a structured framework for researchers and practitioners in various fields, such as healthcare, finance, industry and education, where supervised classification algorithms play a central role in decision-making. In addition, the results highlight the importance of selecting algorithms according to specific needs, and open up promising prospects, including the development of hybrid models and improved real-time data processing.**

Keywords*—Supervised classification; Naïve Bayes; decision tree; Random Forest; k-nearest neighbor; Support Vector Machine; algorithm performance; interpretability

I. INTRODUCTION

In today's era of megadata, supervised classification models play a central role in transforming raw data into actionable information in critical fields such as healthcare, finance and industry. The rise of artificial intelligence (AI) and the increasing availability of data have reinforced the importance of supervised classification as a pillar of machine learning systems. These models are used, for example, to improve medical diagnosis by identifying complex patterns in health data, to detect fraud in finance and to predict preventive maintenance in the manufacturing industry. However, in view of the growing diversity of available models and the varied requirements of practical applications, in-depth benchmarking is needed to guide informed choices and maximize the effectiveness of solutions.

The massive increase in data volumes is accompanied by a proliferation of classification models, each with advantages and limitations depending on the application context. This diversity complicates the choice of the optimal model, requiring an assessment of technical performance, resource requirements and adaptability to specific data types, such as text, images and time series. For example, the prediction of academic success or predictive diagnostics in medicine require models capable of combining accuracy, robustness and speed. A thorough understanding of the trade-offs between these aspects is therefore essential to ensure that models meet performance targets.

Our study responds to a pressing need for synthetic and comprehensive comparisons of supervised classification models. Existing research often focuses on specific criteria, such as accuracy or algorithmic complexity, without offering a clear overview that takes into account the impact on real applications. In addition, certain gaps persist, notably concerning the adaptability of models to multimodal or unbalanced data, and the management of trade-offs between accuracy and computational resources. By proposing a literature review focusing on these criteria, this study aims to provide a valuable tool for researchers and practitioners, enabling them to select algorithms suited to their specific needs.

Faced with these challenges, the central problem of this study can be formulated as follows: What are the best-performing supervised classification models in terms of accuracy, robustness, computational complexity and adaptability, and how does their performance vary according to data type and application context? To answer this question, an in-depth literature review appeared to be the most appropriate method. This approach makes it possible to pool and analyze a wide range of academic and empirical studies on supervised classification models, thus drawing out clear trends and identifying existing trade-offs.

The methodology adopted is based on a rigorous bibliographical search covering publications from scientific journals, international conferences and recent empirical studies. The selected models are evaluated according to defined criteria, such as prediction accuracy, generalizability, robustness, computation time and adaptability to heterogeneous and missing data. These criteria were chosen for their relevance to the evaluation of practical performance, and for their ability to cover the actual requirements of the systems deployed. Strict

inclusion criteria were applied to guarantee the representativeness and quality of the studies selected.

The results show that some models, such as Random Forest and SVM, stand out for their robustness and accuracy in complex contexts, while models like Naïve Bayes and Decision Tree are appreciated for their interpretability, ease of implementation and speed. Comparative analysis has identified notable trade-offs between accuracy and computational complexity, offering guidelines for optimal model selection according to system constraints.

In conclusion, this study proposes a framework for the selection of supervised classification algorithms according to specific user needs. It also highlights promising avenues of research, such as the exploration of hybrid models combining the strengths of the algorithms compared in order to overcome their limitations. Prospects include the development of models better adapted to the challenges of big data and real-time processing requirements.

The paper is structured into five main sections. Section I is dedicated to the introduction and presents the background, problem and objectives of the study. Section II provides a detailed theoretical framework on supervised classification and discusses the main models, while presenting their current limitations and challenges. This section is subdivided into several parts, including an overview of machine learning, a description of the types of supervised classification algorithms and a discussion of their limitations. Section III describes the methodology adopted, covering algorithm selection, evaluation criteria and data collection. Section IV presents the results of the comparative analysis and offers an in-depth discussion of the observations made, followed by a reflection on the limitations of the results. Finally, Section V concludes the study by summarizing the main contributions and suggesting avenues for future research.

II. Theoretical Framework and Background

A. Introduction to Supervised Classification Models

1) Machine learning-an overview: Machine Learning is a branch of artificial intelligence that aims to equip systems with the ability to discover patterns and relationships within datasets, without being explicitly programmed for each specific task [1][2][3]. By forming models from training data, it enables the automation of complex tasks such as image recognition or the prediction of user behavior [1][3][4]. This field relies on techniques that enable machines to learn from past data or interactions with their environment, without requiring specific programming for each task [5][6]. More broadly, Machine Learning encompasses the analysis, design, development and implementation of methods that enable machines to systematically improve themselves, offering solutions to problems through advanced algorithmic approaches [7].

As Mitchell explains in [1], a learning system is able to improve its performance with experience, i.e. from data. Machine Learning can be divided into three broad categories: supervised, unsupervised and reinforcement learning. This classification is based on the way the model learns from the data.

2) Supervised learning: Supervised Learning is an approach to Machine Learning in which a model is trained on a set of labeled data. This means that each instance of training data is associated with a known label or response. The model uses these examples to learn to predict the label of new instances.

In supervised learning, two main tasks are often distinguished:

- Regression: The model predicts a continuous numerical value as a function of input data (e.g. temperature prediction).
- Classification: the model predicts a category or class from a predefined set (e.g., spam detection in e-mail).

Supervised classification, the focus of our study, is therefore a sub-category of supervised learning. Hastie et al in [2], point out that supervised learning is particularly effective when data is abundant and well-labeled, as it enables the construction of accurate and robust models.

3) Supervised classification: Supervised Classification involves using a set of labeled data to train a model that can predict the class to which new observations belong. This is a key task in many fields, from medicine (e.g. medical diagnosis) to security (intrusion detection) to financial services (fraud detection).

The Classification model learns to distinguish between different classes by identifying the characteristics that separate them. For example, in a dataset containing images of dogs and cats, a classification model learns to spot specific characteristics to correctly identify each category.

The advantages of Supervised Classification include:

High accuracy: when data is reliably labeled.

Adaptability: Models can be applied in a variety of sectors requiring precise categorization.

Transparency: Some algorithms, such as decision trees, allow decisions to be interpreted, making it easier to understand the criteria used.

However, these models generally require large, well-balanced data sets to avoid bias and ensure good generalizability. According to Domingos in [3], it is crucial that the training data represent the future data well for the model to be robust and reliable.

B. Main Types of Classification Models

1) Naïve Bayes: Naïve Bayes is a supervised classification algorithm based on Bayes' theorem and the assumption of conditional independence between data characteristics. This algorithm is particularly appreciated for its simplicity and speed, making it highly effective in automatic natural language processing (NLP) and text classification tasks [8].

Despite its simplifying assumption, Naïve Bayes often performs competitively against more complex algorithms.

Mathematically, the algorithm is based on Bayes' theorem, formulated as follows:

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)} \quad (1)$$

Where $P(C|X)$ represents the probability that an observation X belongs to a class C.To simplify the calculation, Naïve Bayes assumes that features x_i are conditionally independent, allowing the following approximation:

$$P(X|C) = \prod_{i=1}^{n} P(x_i|C) \quad (2)$$

This simplification makes Naïve Bayes particularly fast and undemanding in terms of computational resources.

Historically, this algorithm was introduced in the 1960s as a statistical tool for probabilistic classification. Its adoption in machine learning was reinforced by the work of [8], who demonstrated its effectiveness in contexts where conditional independence is respected.

The algorithm works by estimating the probability of each class for a given observation, then assigning that observation to the class with the highest probability. Naïve Bayes calculates these probabilities using pre-processed training data.

There are several variants of Naïve Bayes adapted to specific types of data. Gaussian Naïve Bayes, for example, is used for continuous data, assuming a normal distribution of features [9]. Multinomial Naïve Bayes is particularly suited to discrete data, such as word occurrences in text documents [2], while Bernoulli Naïve Bayes applies to binary data.

Naïve Bayes has many advantages, not least its simplicity of implementation and speed of execution, even on high-dimensional data. However, it suffers from limitations when features are highly correlated or when the conditional independence assumption is violated, which can reduce its accuracy [8].

The algorithm has applications in various fields. For example, it is used for spam filtering, where it classifies e-mails as junk or not [10]. In sentiment analysis, it is used to determine the polarity (positive or negative) of textual opinions [2]. Finally, it is applied to facial recognition, rapidly identifying patterns based on discrete features [9].

In short, Naïve Bayes is a fundamental tool in the supervised learning toolbox, combining mathematical simplicity with practical efficiency. Its remarkable performance in specific contexts, combined with its speed, explains its widespread adoption in many applications.

2) Decision tree: A decision tree is a supervised classification model structured in the form of a tree, where each node represents a question or test on a feature of the data, each branch corresponds to a possible outcome of that test, and each leaf represents a class or final decision. Decision trees are widely used because of their simplicity and interpretability, enabling users to visualize and understand the decision-making process clearly and explicitly [10].

Decision trees are based on sound mathematical foundations, notably the division criteria used to determine the most discriminating features at each stage. Among these criteria, information gain, used by the ID3 and C4.5 algorithms, measures entropy reduction after a specific division [10]. It is defined by the following formula:

$$Gain(T, X) = Entropy(T) - \sum_{v \in V} \frac{|T_v|}{|T|} . Entropy(T_v) \quad (3)$$

where T is the data set at the current node, X is the feature being tested, and T_v represents the subset of data corresponding to a value v of X. The Gini index, used by the CART algorithm, is another popular criterion, quantifying the homogeneity of classes in the subsets generated. It is defined by :

$$Gini(T) = 1 - \sum_{i=1}^{C} P_i^2 \quad (4)$$

where P_i is the proportion of data belonging to class i [11].

The history of decision trees is marked by significant advances. Ross Quinlan introduced the ID3 algorithm in 1986, one of the first models based on information gain [12]. He later developed C4.5, an extension of ID3 capable of handling continuous data and missing values [10]. At the same time, Leo Breiman proposed the CART (Classification and Regression Trees) algorithm in 1984, which introduced the use of the Gini index and opened up possible applications for regression [11]. These contributions laid the methodological foundations for decision trees, positioning them as an essential tool in supervised learning.

Decision trees work by recursively dividing the data according to the most discriminating characteristics, based on the criteria mentioned. For example, to predict whether a person will buy a product based on their age and income, a decision tree would ask questions such as: “Is the person under 30 ?” or “Is the income over 50,000 ?”. Each answer leads to a specific decision, represented by a leaf in the tree.

Decision trees come in several variants, adapted to specific needs. ID3 (Iterative Dichotomiser 3), proposed by [12], is based on information gain as the dividing criterion. Its extensions, C4.5 and C5.0, introduce the management of continuous data and missing values [10]. Finally, CART, developed by study [11], uses the Gini index and can be applied to classification and regression tasks.

The advantages of decision trees are numerous. They offer a high degree of interpretability, thanks to their explicit and easily visualized structure [10]. They can be adapted to both categorical and continuous data, making them versatile in a variety of contexts [11]. What's more, their speed makes them effective for medium-sized data sets. However, they also have their limitations. Decision trees are prone to overlearning, particularly in the absence of regularization, which can impair their generalization ability [13]. They are also sensitive to variations in the training data, which can affect their stability [2].

In terms of applications, decision trees play a key role in various fields. In bioinformatics, they are used to identify genes associated with disease [10]. In the financial sector, they are used to analyze credit decisions and detect fraud [11]. Finally, in industry, they are applied for quality control and predictive maintenance [9]. Their flexibility and simplicity explain their widespread adoption in sectors requiring clear, explainable decisions.

In conclusion, decision trees represent a robust and intuitive method for solving classification and regression problems. Although they present challenges such as overlearning and data sensitivity, their advantages and wide range of applications make them an indispensable tool in the arsenal of supervised classification algorithms.

3) Random forest: Random Forest is a powerful ensemble algorithm that combines several independent decision trees to improve the robustness and accuracy of predictions. Developed by [14], this algorithm is based on two fundamental principles: bootstrap aggregation and random feature selection [14]. By aggregating the predictions of several trees, Random Forest reduces errors due to variance and increases the stability of results.

Mathematically, each tree is constructed from a bootstrap sample of the training data. At each node, only a random sub-selection of features is evaluated to determine the best split. This approach introduces diversity between trees, improving the generalizability of the model.

Historically, Random Forest was conceived as an improvement on conventional decision trees, in response to their tendency to overlearn. [14] Demonstrated that the introduction of randomness in both data and features makes the model more resistant to noisy data and outliers.

Random Forest works by independently generating a large number of decision trees, then combining their predictions. For a classification task, the predicted class is determined by a majority vote among the trees. In regression, the final prediction is obtained by averaging the tree outputs.

The algorithm has several advantages. It is robust to noisy data, efficiently handles unbalanced classes and requires no prior data normalization. However, its main drawback is its increased complexity, which makes interpretation more difficult than with a single decision tree [13].

Random Forest is used in a variety of fields. In finance, it is applied to credit scoring and fraud detection [14]. In bioinformatics, it is used for feature selection and classification of genomic data [9]. In the energy sector, it is used to predict energy consumption as a function of various factors [13].

In conclusion, Random Forest is a flexible, high-performance method for a wide range of classification and regression tasks. Its robustness and precision make it an indispensable tool in complex, noisy environments.

4) K-Nearest Neighbor (KNN): K-Nearest Neighbor (KNN) is a supervised classification algorithm based on the concept of proximity in a multidimensional space. It classifies an observation according to the majority class among its k nearest neighbors. Introduced by study [15], KNN is one of the simplest and most intuitive algorithms in supervised learning [15].

In mathematical terms, KNN relies on a distance measure to assess the proximity between observations. The Euclidean distance is the most commonly used and is defined by:

$$d(x, y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad (5)$$

where x_i and y_i represent the characteristics of points x and y. For a new point, KNN identifies the k nearest neighbors, then determines the majority class among them.

Historically, KNN has gained in popularity thanks to its simplicity and adaptability to different types of data. However, it is particularly sensitive to the choice of k, which directly influences its performance. Too small a k makes the model sensitive to noise, while too large a k can dilute important features [2].

KNN works as follows: for each new observation, the algorithm identifies the k closest observations in the training data, using a predefined distance measure. It then assigns the majority class among these neighbors to the observation in question.

KNN offers several advantages, not least its simplicity and effectiveness on small or well-separated data sets. However, it is computationally expensive for large datasets, as it requires each new observation to be compared with the training dataset. In addition, it is sensitive to irrelevant or redundant features, which can affect its accuracy [2].

KNN has many applications. In product recommendation, it identifies similar products based on user preferences. In image recognition, it classifies images based on distances in pixel space. In healthcare, KNN is used to diagnose diseases based on biological characteristics [2].

In summary, KNN is a versatile and efficient algorithm for classification and regression tasks. Despite its limitations in terms of computational complexity, its simplicity and adaptability make it a preferred choice for applications requiring proximity-based classification.

5) Support Vector Machine (SVM): The Support Vector Machine (SVM) is a supervised classification algorithm designed to maximize the margin between classes in a multidimensional space. Developed by study [16], SVM revolutionized classification methods by introducing the idea of kernels to project data into higher-dimensional spaces, thus facilitating their linear separation [16].

$$\min_{\omega,b} \frac{1}{2}\|\omega\|^2 \quad (6)$$

under the constraints :

$$y_i(\omega^T x_i + b) \geq 1, \forall i \quad (7)$$

where ω is the weight vector, b the bias, x_i the observations and y_i their respective labels. By maximizing the margin

between the points closest to the classes, called support vectors, SVM guarantees better generalization [17].

The use of kernels enables SVM to handle non-linear problems. Common kernels include linear, polynomial and radial basis function (RBF). These functions transform the data into a higher-dimensional space, where it becomes linearly separable [16].

SVM works by constructing an optimal hyperplane that maximizes the margin between classes. For non-linear data, SVM uses kernels to perform an implicit transformation into a higher-dimensional space. In this way, data are optimally separated, minimizing classification errors.

SVM is particularly efficient for small to medium-sized datasets, but its computational cost can become prohibitive for large databases. In addition, it requires careful selection of hyperparameters, such as the C parameter (controlling the trade-off between maximum margin and error) and the choice of kernel [17].

SVM applications cover a wide range of fields. In bioinformatics, it is used to classify genes and predict diseases. In the financial sector, it analyzes trends and predicts market behavior. In computer vision, SVM is applied to facial and object recognition [2].

In conclusion, SVM is a powerful and flexible algorithm, capable of processing both linear and non-linear data. Although it requires expertise to optimize its parameters, its accuracy and ability to handle complex problems make it a preferred choice for many applications.

C. Current Limitations and Challenges

Despite their many applications and successes, supervised classification algorithms have inherent limitations and challenges that impact their adoption in certain contexts. These limitations vary from algorithm to algorithm and are influenced by factors such as the nature of the data, available resources, and the specific requirements of application domains.

1) Dependence on data quality: The performance of supervised classification algorithms is highly dependent on the quality of the training data. Noisy, unbalanced or incomplete data can lead to bias and reduced accuracy [2][18]. For example, Decision Trees are prone to overfitting when training data contains anomalies or extreme values [10].

2) Scalability problem: Algorithms such as KNN and SVM can be inefficient on large datasets due to their algorithmic complexity [9][14]. KNN, with its O(n) complexity, becomes impractical for databases containing millions of points [15].

3) Lack of interpretability: Some algorithms, such as Random Forest, produce complex models that are difficult for non-expert users to interpret [14][19]. Although Random Forest offers high accuracy, its inner workings are often described as “black box”, limiting its adoption in sensitive fields such as medicine [19].

4) Sensitivity to overfitting: Models such as Decision Trees tend to overfit on training data, losing their ability to generalize on new data [10][2]. An unregularized Decision Tree may provide perfect results on training data but fail on test sets [13].

5) Resources and computing time: Algorithms like SVM and Random Forest require significant resources for training, which can be prohibitive in resource-constrained environments [9][16]. SVM with nonlinear kernels can require several hours to train on large datasets [16].

6) Adaptability problem: Some models, such as Naïve Bayes, assume feature independence, which limits their performance on datasets where features are highly correlated [8][9]. In textual datasets, Naïve Bayes' conditional independence assumption can lead to a significant loss of accuracy [8].

7) Hyperparameter dependence: Algorithm performance often depends on the fine-tuning of hyperparameters, which requires in-depth expertise and resources for cross-validation [2][16]. SVM efficiency is strongly influenced by the choice of kernel, C, and γ [2].

8) Generalization and robustness: Models sometimes lack robustness in the face of unbalanced or noisy data, which can lead to inconsistent performance [20][21]. Although Random Forests are robust in many cases, their performance can plummet when minority classes are severely under-represented [20].

9) Ethical and regulatory challenges: The use of models in sensitive areas (health, finance) raises ethical issues related to fairness and transparency [19]. Biases introduced in training data can lead to discriminatory decisions in classification systems [19].

These limitations and challenges underline the need to tailor supervised classification algorithms to the specifics of applications. Future research should focus on developing models that are more interpretable, scalable, and robust to data variations. At the same time, solutions such as model hybridization or the integration of unsupervised approaches can help overcome some of these challenges.

III. METHODOLOGY

The aim of this work is to analyze and compare supervised classification algorithms and ensure a comprehensive and balanced evaluation based on well-defined criteria and a rigorous literature review. The methodology includes algorithm selection, definition of evaluation criteria, data collection and organization of the comparative synthesis.

A. Algorithm Selection

Five algorithms widely recognized in the scientific and industrial literature were selected: Naïve Bayes, Decision Tree, Random Forest, K-Nearest Neighbor (KNN) and Support Vector Machine (SVM).

This choice is justified by:

- Representativeness of approaches: These algorithms cover a wide range of methods (probabilistic, tree-based, instance-based, etc.).

- Popularity and relevance: Their widespread adoption in industry and research testifies to their effectiveness in a variety of contexts [14], [16].
- Richness of available studies: These algorithms have been extensively documented, enabling in-depth comparison based on reliable experimental data. Naïve Bayes for text classification and spam detection [8]. Random Forest for fraud detection and medical diagnostics [14].

B. Criteria for Comparing Classification Algorithms

The comparison of classification algorithms requires rigorously defined criteria to assess their performance and suitability for specific tasks. This subsection presents the criteria and sub-criteria used, explains their importance, and justifies their selection.

1) Presentation of the criteria

- Criteria for comparing supervised classification algorithms play a key role in assessing their effectiveness, performance and adaptability to different contexts. These criteria, carefully defined and justified, cover the essential dimensions for ensuring a rigorous and scientifically valid analysis.
- Prediction accuracy is one of the most fundamental criteria, measuring the percentage of observations correctly classified by a model. It is calculated as the ratio between the number of correct predictions and the total number of predictions [2]. This criterion is crucial for applications where errors could have serious consequences, such as in medicine or finance. It directly reflects the reliability of a model and its suitability for accuracy requirements in critical environments.
- Robustness represents the ability of an algorithm to maintain its performance despite noisy or disturbed data [13]. In real-world environments where data is often imperfect, this criterion ensures that the algorithm remains reliable, even in the presence of errors or anomalies in the data. This is particularly important in fields such as bioinformatics or facial recognition.
- The tendency to overfitting assesses whether a model is able to avoid too much adaptation to training data, which would compromise its ability to generalize on new data [2]. Algorithms with a high risk of overfitting often require regularization techniques to guarantee stable performance. This criterion is essential in applications requiring the ability to generalize over varied data sets.
- Generalizability is closely linked to the tendency to overfitting, and measures the extent to which an algorithm can successfully predict on unseen data [22]. This criterion is essential for assessing the durability of model performance in dynamic and unpredictable environments.
- Training time refers to the time required to build a model from training data [2]. In contexts where models need to be updated frequently, such as real-time recommendation systems, this criterion helps to identify algorithms suited to strict time constraints.
- Prediction time corresponds to the time required to provide a prediction on new data [2]. This criterion is essential in applications where responsiveness is crucial, such as autonomous vehicles or industrial control systems. It ensures that the algorithm can respond quickly to real-time requests.
- Memory requirement measures the amount of memory needed to store the model and make predictions [17]. In resource-constrained environments, such as embedded devices, this criterion enables the selection of memory-efficient algorithms.
- Algorithmic complexity evaluates the computational cost of an algorithm in terms of time and space, often expressed in O notation (Hastie et al., 2009). This criterion is crucial in determining whether an algorithm can be used efficiently with large datasets or in resource-constrained environments.
- Ease of implementation examines the simplicity with which an algorithm can be integrated into an existing system [10]. An easy-to-deploy model reduces the costs and time associated with development, which is particularly advantageous in projects requiring rapid integration.
- Model interpretability refers to a model's ability to provide understandable explanations for its predictions [19]. This criterion is essential in regulated sectors, such as healthcare or finance, where transparency of decisions is paramount in building user trust.
- The level of expertise required reflects the skills needed to configure and use an algorithm effectively [3]. This criterion is important for organizations with limited human resources in technical expertise, as it can influence the ease of model adoption.
- Available documentation and libraries also play a key role in algorithm implementation [23]. Comprehensive documentation and well-supported libraries simplify the learning process and enable faster adoption of algorithms.
- Adaptability to multimodal data assesses a model's ability to simultaneously process different types of data, such as text, images or audio signals [21]. This criterion is crucial for modern applications such as voice assistants or multimodal recognition.
- Adaptability to unbalanced data measures an algorithm's ability to efficiently handle datasets where certain classes are under-represented [20]. This criterion is essential in fields such as fraud detection or the prediction of rare diseases, where the scarcity of cases of interest complicates model training.

- Handling missing data is an important criterion for assessing an algorithm's ability to operate in environments with incomplete data [18]. Models capable of tolerating missing values are particularly valuable in applications where complete data is rarely available.
- Dimensionality reduction is a criterion that measures a model's ability to identify and use the most important features while reducing noise [24]. This improves the efficiency of algorithms and their ability to handle complex data sets.
- Application domains reflect the usefulness of algorithms in a variety of sectors, such as finance or healthcare, where solutions like Random Forest have demonstrated their effectiveness in fraud detection and genomic analysis [14]. This criterion is crucial for assessing the relevance of algorithms in specific contexts.
- Impact on practical decisions assesses the direct contribution of algorithms to decision-making. For example, Decision Trees are valued for their interpretability, facilitating reliable medical diagnoses [10]. This criterion is essential in determining how an algorithm can transform critical decision-making processes.
- Integration into existing systems measures the ease with which algorithms can be implemented. Naïve Bayes are simple to deploy and easily adaptable to tools such as spam filtering systems [3]. Conversely, SVMs, with their complexity, pose integration challenges [16], underlining the importance of this criterion in assessing their practical applicability.
- Implementation cost, finally, evaluates the resources required to operate an algorithm. Naïve Bayes, for example, are cost-effective, unlike SVMs, which require greater resources due to their algorithmic complexity [2]. This criterion is essential for judging the suitability of algorithms for real system constraints.

2) Distribution of evaluation criteria

- Technical performance: Prediction accuracy, robustness, tendency to overfitting and generalizability.
- Computational efficiency: Computation time for training and prediction, memory requirements and algorithmic complexity.
- Practicality and Ease of Use: Ease of implementation, Interpretability of models, Level of expertise required and Documentation and libraries available.
- Data compatibility: Adaptability to data (multimodal, unbalanced), Handling of missing data and Reduction of dimensionality.
- Applicability and Practical Impact: Application domains, Impact on practical decisions, Integration into existing systems and Implementation cost.

These dimensions cover the essential aspects of algorithms, ensuring a balanced assessment between their raw performance, practicality, and adaptability to real-life use cases [2] [21].

3) Data collection

a) Publication selection criteria

- Period: Publications from 2000 to 2025 to include recent advances while maintaining a historical perspective.
- Types of publication: Papers in indexed journals (Elsevier, Springer, IEEE), Proceedings of major conferences (NeurIPS, ICML, AISTATS) and relevant industrial technical reports.
- Context of studies: Include work covering various fields of application (health, finance, education, etc.).

b) Research process

- Databases searched: Google Scholar, PubMed, IEEE Xplore, Scopus.
- Screening: The selected studies had to provide detailed experimental results and quantitative comparisons between several algorithms.

c) Validation of results

- Each publication is reviewed in two stages: Analysis of the abstract and introduction to confirm relevance, and reading of the methodology and results sections to ensure data quality.

d) Data synthesis

- Thematic organization: Publications are grouped by application (healthcare, finance, industry, etc.) and evaluation criteria (accuracy, robustness, etc.).
- Metrics extraction: Experimental results (accuracy, computation time) are extracted to feed comparative analyses.

4) Organization of the comparative summary

a) Results format: Synthetic tables: Presentation of criteria in tabular form for quick comparison of algorithms.

b) Qualitative analysis: Textual explanation of algorithm strengths and weaknesses, contextualized by extracted results.

c) Cross-validation: Comparison of experimental results between different publications to reinforce reliability. Narrative analysis to incorporate variations due to specific study contexts. Fig. 1 shows the comparative study workflow.

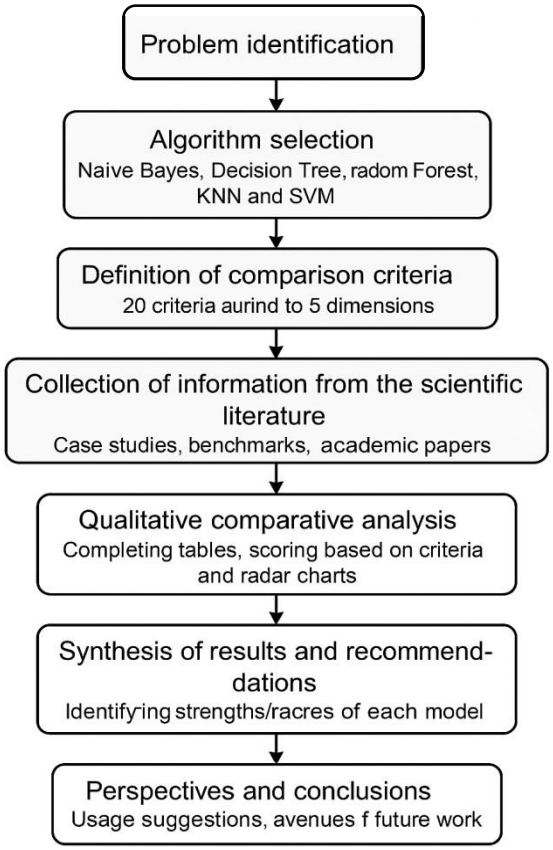


Fig. 1. The comparative study workflow shows the overall study process.

IV. Results and Discussion

A. Presentation of Results

The results are organized according to the dimensions defined in the methodology: technical performance, computational efficiency, practicality, compatibility with data, and practical applicability.

1) Technical performance: This table evaluates the accuracy, robustness to perturbations, tendency to overfitting, and generalizability of the models. The qualitative scores are normalized on a scale of 5 and accompanied by accuracy ranges reported in the literature. Scientific references are provided for each criterion to support the evaluations. Random Forest (RF) and Support Vector Machine (SVM) models stand out overall for their high performance, while Naive Bayes (NB) shows more variable performance depending on the criteria.

This figure illustrates the comparative performance of the five supervised classification algorithms (Naive Bayes, Decision Tree, Random Forest, SVM, and KNN) based on the average scores assigned to each according to the five dimensions defined in the study: technical performance, computational efficiency, ease of use, data compatibility, and practical applicability. The colors of the bars represent the different dimensions of analysis (e.g., yellow = performance, orange = efficiency, red = ease of use, blue = data compatibility, etc.). Each score ranges from 1 (low) to 5 (excellent). We can see that Random Forest and SVM achieve the highest scores overall, while Naive Bayes and KNN show more mixed results depending on the criteria.

TABLE I. Comparison of Algorithms Based on Technical Performance Criteria

Algorithm	Accuracy (%) Yellow (Y)	Robustness Orange(O)	Tendency to overfitting Red(R)	Generalizability Blue (B)
NB	3/5 45 – 98 [25][26][27] [28][29]	2/5 [40]	4/5 [45]	3/5 [9]
DT	3/5 64 – 95 [25][30][31] [32][33]	3/5 [41]	2/5 [46]	3/5 [14]
RF	5/5 71 – 98 [30][31][33] [34][35]	5/5 [42]	4/5 [41]	5/5 [49]
KNN	3/5 64 – 98 [26][36][36] [38][39]	2/5 [43]	2/5 [47]	4/5 [50]
SVM	5/5 70 – 97 [27][28][29] [35][38]	4/5 [44]	4/5 [48]	5/5 [9]

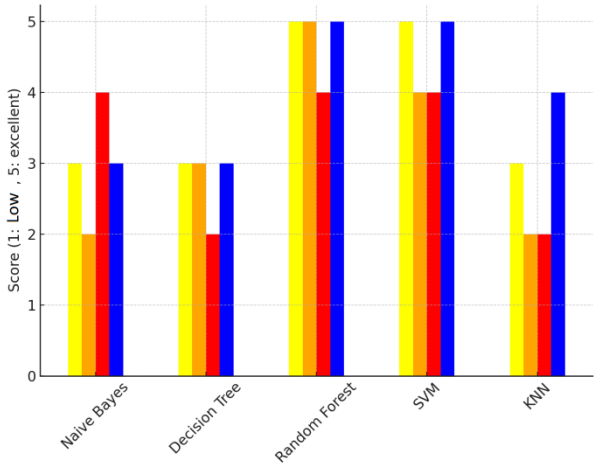


Fig. 2. Comparison of algorithms based on technical performance criteria.

2) Computational efficiency: This table compares the five algorithms studied according to four key aspects of computational efficiency. Each criterion is evaluated on a scale of 1 to 5. The results highlight the superiority of Naive Bayes across all criteria, with perfect scores (5/5), while Random Forest and SVM show compromises between accuracy and computational cost. The bibliographic references associated with each score are also indicated to ensure the traceability of the evaluations.

TABLE II. COMPARISON OF ALGORITHMS BASED ON COMPUTATIONAL EFFICIENCY

Algorithm	Time for training (Y)	Time for prediction (B)	Memory requirements (O)	Algorithmic complexity (R)
NB	5/5 [45]	5/5 [3]	5/5 [2]	5/5 [45]
DT	4/5 [47]	4/5 [47]	4/5 [2]	4/5 [41]
RF	2/5 [49]	3/5 [41]	3/5 [42]	2/5 [49]
KNN	3/5 [51]	1/5 [43]	2/5 [43]	3/5 [51]
SVM	2/5 [52]	3/5 [48]	3/5 [48]	2/5 [52]

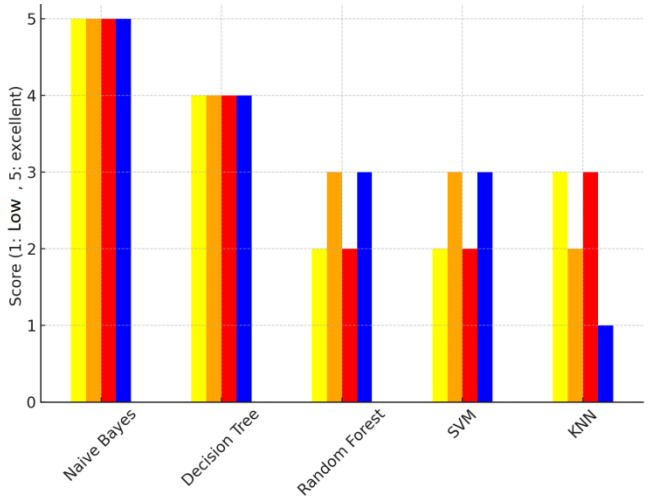


Fig. 3. Comparison of algorithms based on computational efficiency.

This figure provides a comparative visual representation of the five algorithms studied in Table II, according to the following criteria: Time for training (Yellow), Memory requirements (Orange), Algorithmic complexity (Red), and Time for prediction (Blue). Each criterion is rated on a scale of 1 to 5.

3) Practicality and ease of use: This table evaluates the five supervised classification algorithms (Naive Bayes, Decision Tree, Random Forest, k-NN, SVM) according to four practical criteria: ease of implementation, interpretability, level of expertise required, and availability of documentation and associated libraries. Scores are rated on a scale of 1 to 5.

A simplified visualization of the results in Table III, this figure compares the performance of the five algorithms according to four criteria: Ease of implementation (yellow), Interpretability (orange), Level of expertise required (red), and Documentation and libraries available (blue). Scores are given on a scale of 5 to allow for quick comparative reading.

4) Data compatibility: This table evaluates the five algorithms according to four compatibility sub-criteria: Adaptability to multimodal data, Management of missing data, Dimensionality reduction, and Adaptability to unbalanced data. Each criterion is scored out of 5, and the associated bibliographic references are indicated in brackets. These scores illustrate the extent to which each model is able to adapt to the diversity and imperfections of the data used in supervised classification.

TABLE III. COMPARISON OF ALGORITHMS BASED ON PRACTICALITY AND EASE OF USE

Algorithm	Ease of implementation (Y)	Interpretability (O)	Level of expertise require (R)	Documentation and libraries available (B)
NB	5/5 [45]	3/5 [3]	5/5 [45]	5/5 [9]
DT	4/5 [51]	5/5 [53]	4/5 [48]	5/5 [23]
RF	3/5 [41]	2/5 [49]	3/5 [52]	5/5 [9]
KNN	5/5 [51]	2/5 [43]	5/5 [51]	5/5 [23]
SVM	2/5 [48]	2/5 [54]	2/5 [48]	5/5 [9]

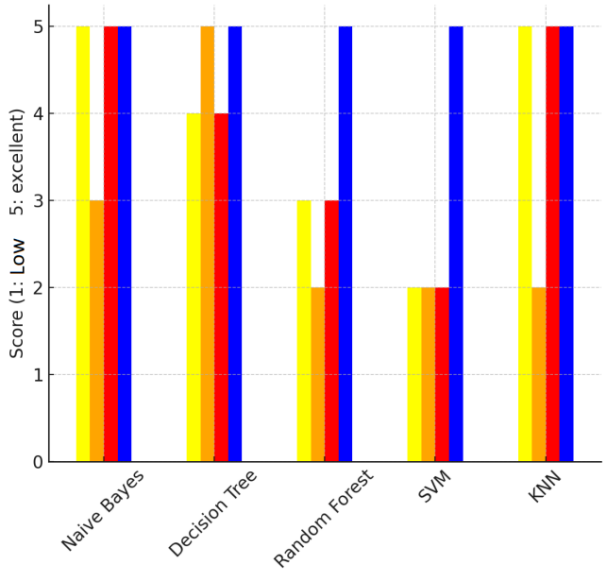


Fig. 4. Comparison of algorithms based on practicality and ease of use.

TABLE IV. COMPARISON OF ALGORITHMS BASED ON COMPATIBILITY WITH DATA

Algorithm	Adaptability to multimodal data (Y)	Handling of missing data (O)	Reduction of dimensionality (R)	Adaptability to unbalanced data (B)
NB	3/5 [55]	3/5 [18]	4/5 [24]	2/5 [55]
DT	4/5 [47]	4/5 [41]	3/5 [47]	3/5 [54]
RF	5/5 [49]	5/5 [42]	4/5 [49]	4/5 [49]
KNN	3/5 [51]	2/5 [43]	3/5 [51]	2/5 [43]
SVM	4/5 [54]	2/5 [48]	5/5 [48]	4/5 [52]

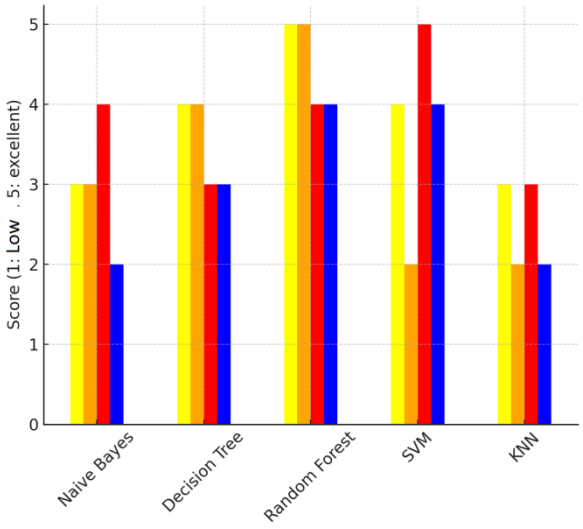


Fig. 5. Comparison of algorithms based on compatibility with data.

Summary graph based on data from Table IV, showing the scores assigned to algorithms on four criteria: Adaptability to multimodal data (yellow), Handling of missing data (orange), Reduction of dimensionality (red), and Adaptability to unbalanced data (blue). This visualization facilitates comparative interpretation between models.

5) Applicability and practical impact: This table presents a comparative evaluation of five supervised classification algorithms—Naive Bayes (NB), Decision Tree (DT), Random Forest (RF), k-Nearest Neighbors (k-NN) [37], and Support Vector Machine (SVM)—based on four criteria related to their applicability in real-world contexts. The criteria studied include: (1) diversity of application domains, (2) impact on practical decision-making, (3) degree of integration into existing systems, and (4) implementation cost. Each criterion is rated on a scale of 1 to 5, where 5 indicates the best performance. The scores are accompanied by bibliographic references [45], [2], [14], etc., attesting to the origin of the evaluations. This summary aims to guide practitioners in choosing an algorithm based on concrete constraints, such as ease of implementation, compatibility with existing systems, or decision-making scope. It thus complements purely technical or statistical comparisons by integrating the dimension of operational applicability.

This visualization illustrates the comparative scores from Table V for the four criteria selected: Application domains (yellow), Impact on practical decisions (orange), Integration into existing systems (red), and Implementation cost (blue). It highlights the relative strengths of each algorithm through a clear representation.

B. Discussion

Quantitative analysis of supervised classification algorithms reveals clear but complementary differences in their technical, operational, and practical properties. These results can be interpreted in several ways depending on the deployment objectives and context of use.

TABLE V. COMPARISON OF ALGORITHMS ACCORDING TO APPLICABILITY AND PRACTICAL IMPACT

Algorithm	Application domains (Y)	Impact on practical decisions (O)	Integration into existing systems (R)	Implementation cost (B)
NB	4/5 [45]	3/5 [2]	4/5 [45]	5/5 [9]
DT	4/5 [41]	4/5 [14]	4/5 [48]	4/5 [41]
RF	5/5 [49]	5/5 [49]	4/5 [42]	3/5 [49]
KNN	3/5 [43]	3/5 [51]	3/5 [51]	5/5 [51]
SVM	5/5 [56]	4/5 [54]	3/5 [52]	3/5 [48]

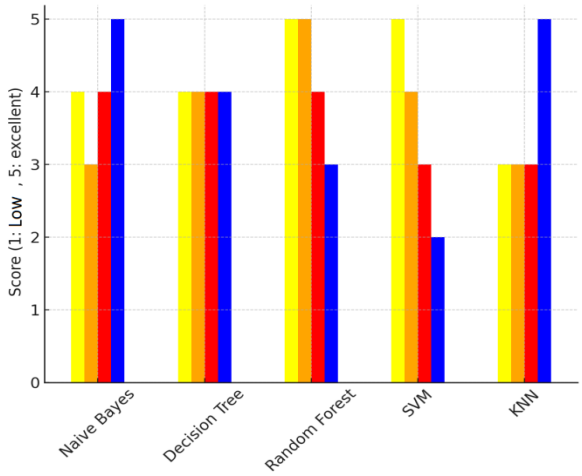


Fig. 6. Comparison of algorithms according to applicability and practical impact.

The Random Forest and SVM algorithms clearly stand out in terms of accuracy, generalization ability, and robustness (Table I, Fig. 2). Their high scores confirm their ability to handle complex problems requiring reliable prediction, such as in medical imaging or cybersecurity. Conversely, although Naive Bayes has an elegant probabilistic structure, its simplification of inter-variable dependencies limits its overall accuracy, particularly in highly correlated contexts. Decision Tree shows moderate performance, while KNN suffers from high sensitivity to noise and overfitting.

Naive Bayes and, to a lesser extent, Decision Tree, are notable for their computational lightness (Table II, Fig. 3). These methods are appropriate in real-time processing contexts or on machines with limited resources. In contrast, Random Forest and SVM, although powerful, have a significant computational cost, both in training and prediction. KNN, although without a learning phase, is slowed down at the time of classification due to the need to scan the entire dataset.

Naive Bayes and KNN stand out as the most accessible, requiring little expertise and being very easy to implement (Table III, Fig. 4). Decision Tree combines this accessibility with excellent interpretability, making it a popular model for decision support systems. On the other hand, SVM and Random Forest are often perceived as “black boxes” requiring

more sophisticated configuration and increased expertise, which can limit their adoption in constrained industrial contexts.

The results show that Random Forest is the most versatile, effectively handling missing data, class imbalances, and dimensionality reduction. SVM follows closely behind, provided rigorous preprocessing is performed (Table IV, Fig. 5). Naive Bayes and KNN, on the other hand, are sensitive to data quality and structure. The former assumes conditional independence of variables, while the latter suffers from the effects of dimensionality and noise.

Random Forest and Decision Tree are very attractive from an application standpoint, with broad sectoral adaptability and a concrete impact on decision-making (Table V, Fig. 6). Naive Bayes is also valued for its low cost and ease of integration. On the other hand, SVM remains more expensive to implement, although it offers high performance. Finally, KNN remains unpopular in production environments due to its limited scalability.

C. Limitations of the Results

Although this comparative review provides an in-depth summary of the literature, several limitations should be noted:

1) Lack of empirical validation: The scores are based on bibliographic analyses and do not reflect direct experiments on specific datasets. An empirical study would have allowed for more contextual calibration of the assessments.

2) Variability of application contexts: Algorithm performance can vary significantly depending on the domain (healthcare, finance, language processing, etc.). As this study is general in nature, it does not capture all the nuances specific to each sector.

3) Implementation changes: The performance and practicality of algorithms can be influenced by the software libraries used, which evolve rapidly. Results may therefore change in the short term.

4) Equal weighting of criteria: Aggregating scores assumes uniform weighting of the 20 criteria, which does not necessarily reflect the actual importance of each criterion depending on the context of use.

V. Conclusion

This comparative study, based on an in-depth literature review, evaluated five major supervised classification algorithms—Naive Bayes, Decision Tree, Random Forest, Support Vector Machine (SVM), and K-Nearest Neighbor (KNN)—across twenty criteria divided into five dimensions: technical performance, computational efficiency, practicality, data compatibility, and applicability.

The results show that Random Forest offers an excellent compromise between accuracy, robustness, and applicability, making it a preferred choice for high-stakes use cases. Naive Bayes, on the other hand, stands out for its simplicity, speed, and low implementation cost, making it ideal for resource-constrained environments. Decision Tree occupies a middle ground, combining interpretability and good performance. On the other hand, SVM, despite its remarkable technical performance, remains difficult to implement and integrate into real systems. Finally, KNN has significant limitations in terms of scalability and robustness, although it remains useful for simple and exploratory cases.

This study highlights the importance of a contextualized multi-criteria assessment in the choice of classification algorithms. No model is universally applicable; each algorithm has strengths and weaknesses that must be weighed according to the specific constraints of the problem, the domain, and the technical environment.

Future work could expand on this bibliographic analysis through rigorous experimental validation, applying the five classification algorithms to real-world datasets from various fields such as education, healthcare, and finance. Such an extension would allow theoretical observations to be compared with empirical performance and recommendations for use to be refined according to the application context. In addition, a promising avenue lies in integrating recent advances in AutoML to automate the selection, configuration, and optimization of models based on the nature of the data. Exploring hybrid learning techniques (e.g., combining multiple algorithms via ensembles or mixed architectures) could also improve robustness and predictive performance. Finally, the use of supervised deep learning approaches, in particular deep neural models or pre-trained neural networks (e.g., BERT, LSTM, CNN), would represent a step forward in the processing of more complex data (text, images, etc.). A complementary benchmark could thus be considered to compare classical models and neural architectures on broader common criteria (training time, ability to generalize, sensitivity to imbalances, etc.).

References

[1] Mitchell, T. M. (1997). Machine Learning. McGraw-Hill.

[2] Hastie, T., Tibshirani, R., Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer.

[3] Domingos, P. (2012). A Few Useful Things to Know About Machine Learning. Communications of the ACM, 55(10), 78-87.

[4] LeCun, Y., Bengio, Y., Hinton, G. (2015). Deep Learning. Nature, 521(7553), 436-444.

[5] Murphy, K. P. (2012). Machine Learning: A Probabilistic Perspective. The MIT Press.

[6] Mondal, B. (2020). Artificial Intelligence: State of the Art. In: Balas, V., Kumar, R., Srivastava, R. (eds), Recent Trends and Advances in Artificial Intelligence and Internet of Things. Intelligent Systems Reference Library, vol 172. Springer, Cham.

[7] Erraissi, A., Belangour, A., Tragha, A. (2017). A Comparative Study of Hadoop-based Big Data Architectures. International Journal of Web Applications, 9(4), 129-137.

[8] Domingos, P., Pazzani, M. (1997). On the Optimality of the Simple Bayesian Classifier under Zero-One Loss. Machine Learning, 29(2-3), 103-130.

[9] Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. O'Reilly Media.

[10] Quinlan, J. R. (1993). C4.5: Programs for Machine Learning. Morgan Kaufmann.

[11] Breiman, L., Friedman, J. H., Olshen, R. A., Stone, C. J. (1984). Classification and Regression Trees. Chapman & Hall/CRC.

[12] Quinlan, J. R. (1986). Induction of Decision Trees. Machine Learning, 1(1), 81-106.

[13] Dietterich, T. G. (2000). Ensemble Methods in Machine Learning. Proceedings of the International Workshop on Multiple Classifier Systems.

[14] Breiman, L. (2001). Random Forests. Machine Learning, 45(1), 5-32.

[15] Cover, T. M., Hart, P. E. (1967). Nearest Neighbor Pattern Classification. IEEE Transactions on Information Theory, 13(1), 21-27.

[16] Vapnik, V. N. (1995). The Nature of Statistical Learning Theory. Springer.

[17] Schölkopf, B., Smola, A. J. (2001). Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press.

[18] Little, R. J. A., Rubin, D. B. (2020). Statistical Analysis with Missing Data. Wiley.

[19] Lipton, Z. C. (2016). The Mythos of Model Interpretability. arXiv preprint arXiv:1606.03490.

[20] He, H., Garcia, E. A. (2009). Learning from Imbalanced Data. IEEE Transactions on Knowledge and Data Engineering, 21(9), 1263-1284.

[21] Zhou, Z. H., Feng, J. (2020). Deep Forest: Towards an Alternative to Deep Neural Networks. Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI).

[22] Caruana, R., Niculescu-Mizil, A. (2006). An Empirical Comparison of Supervised Learning Algorithms. Proceedings of the 23rd International Conference on Machine Learning (ICML).

[23] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825-2830.

[24] Guyon, I., Elisseeff, A. (2003). An Introduction to Variable and Feature Selection. Journal of Machine Learning Research, 3, 1157-1182.

[25] Hafidz, H., & Fakhriza, M. (2024). Comparison of Naive Bayes Algorithms and Decision Tree for Classifying Hero Fighter Items in the Mobile Legends. Journal of Applied Science, Engineering, Technology, and Education, 6(2), 127-142.

[26] Buhori, K. (2024). Comparative Analysis of KNN and Neavy Bayes Algorithms in Socio-Economic Data Classification in Indonesia. Digital Zone: Jurnal Teknologi Informasi Dan Komunikasi, 15(2), 222–232.

[27] Sari, P. R., Indah, D. R., Rasywir, E., Firdaus, Mgs. A., & Athalina, G. (2024). Comparison of Naive Bayes and SVM Algorithms for Sentiment Analysis of PUBG Mobile on Google Play Store. Jurnal Sistem Informasi, 13(6), 2767.

[28] Azhar, A., Masruroh, S. U., Wardhani, L. K., & Okfalisa, O. (2023). Performance comparison of the Naive Bayes algorithm and the k-NN lexicon approach on Twitter media sentiment analysis. Science, Technology, and Communication Journal, 3(2), 35-40.

[29] Wibawa, A. P., Kurniawan, A. C., Murti, D. M. P., Adiperkasa, R. P., Putra, S. M., Kurniawan, S. A., & Nugraha, Y. R. (2019). Naïve Bayes Classifier for Journal Quartile Classification. International Journal of Recent Contributions from Engineering, Science & IT (iJES), 7(2), 91–99.

[30] Kinasih, A., Handayani, A., Ardiansah, J., Damanhuri, N. (2024). Comparative analysis of decision tree and random forest classifiers for structured data classification in machine learning. Science in Information Technology Letters, 5, 13-24.

[31] Esmaily, H., Tayefi, M., Doosti, H., Ghayour-Mobarhan, M., Nezami, H., Amirabadizadeh, A. (2018). A Comparison between Decision Tree and Random Forest in Determining the Risk Factors Associated with Type 2 Diabetes. J Res Health Sci, 18(2), 412.

[32] Mohammadi-Pirouz, Z., Hajian-Tilaki, K., Sadeghi Haddat-Zavareh, M., et al. (2024). Development of decision tree classification algorithms in predicting mortality of COVID-19 patients. Int J Emerg Med, 17, 126.

[33] Mitschek, M., Esquivel, R. (2023). A Comparative Analysis of Decision Tree Classification Algorithms for Blended Learning Analytics in WEKA. Celt, 23, 321-334.

[34] Gangadhar, C., Roy, P., Kumar, R., Ramesh, J., Ravikanth, S., Akhila, N. (2025). Wearable Sensor-based Fall Detection for Elderly Care Using Ensemble Machine Learning Techniques. Measurement Sensors, 101870.

[35] Afuan, L., & Isnanto, R. (2024). Enhanced Fall Detection using Optimized Random Forest Classifier on Wearable Sensor Data. Journal of Applied Data Sciences, 6(1), 213-224.

[36] Madi, S., Baba-Ali, A.R. (2024). A new hybrid incremental learning system for an enhanced KNN algorithm (hoKNN). Evolving Systems, 15, 1001–1019.

[37] Uddin, S., Haque, I., Lu, H., et al. (2022). Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction. Sci Rep, 12, 6256.

[38] Shdefat, A.Y., Mostafa, N., Al-Arnaout, Z., et al. (2024). Optimizing HAR Systems: Comparative Analysis of Enhanced SVM and k-NN Classifiers. Int J Comput Intell Syst, 17, 150.

[39] Çetin, A., Büyüklü, A. (2024). A new approach to K-nearest neighbors distance metrics on sovereign country credit rating. Kuwait Journal of Science, 52(1).

[40] Sanjay, K. D., Arya, C. (2016). Automatic Text Classification in Information Retrieval: A Survey. ICTCS '16, ACM, Article 131.

[41] Goyal, G., Karwal, S., Garg, R. (2023). Performance Comparison of Dysgraphia Detection Algorithms. 2022 OPJU International Technology Conference on Emerging Technologies for Sustainable Development (OTCON), IEEE.

[42] Kumar, S., Laxkar, D., Adhikari, S., Vijayarajan, V. (2017). Assessment of various supervised learning algorithms using different performance metrics. IOP Conf. Series: Materials Science and Engineering, 263, 042087.

[43] Siraj-Ud-Doulah, M., Alam, M. (2020). Performance Evaluation of Machine Learning Algorithms in Ecological Dataset. Int. J. of Applied Mathematics and Machine Learning.

[44] Chowdhury, M. S. (2023). Comparison of Accuracy and Reliability of Random Forest, SVM, ANN and Maximum Likelihood method in Land use/cover Classification. Environmental Challenges.

[45] Angula, T., Hashiyana, V. (2023). Detection of Structured Query Language Injection Attacks Using Machine Learning Techniques. Int. J. of Computer Science and Information Technology.

[46] Darde, S., Wagh, S. (2024). To Study the Comparative Analysis of Classification Algorithms for Heart Stroke Prediction. Int. J. of Advanced Research in Science, Communication and Technology.

[47] El-Habil, A. (2014). Evaluation of Data Mining Classification Models. IUG Journal of Natural and Engineering Studies, 151–165.

[48] Rodrigues, I., Parayil, A., Shetty, T., Mirza, I. (2020). Use of LDA, KNN, CART, RF, Gaussian NB, SVM to Predict Admission for PG Courses. SSRN.

[49] Das, A., Choudhury, D., Sen, A. (2024). A collaborative empirical analysis on ML-based disease prediction in healthcare. Int. J. of Inf. Technol., 16, 261–270.

[50] Skowronski, L., de Moraes, P.M., de Moraes, M.L.T., et al. (2021). Supervised learning algorithms in the classification of plant populations. Braz. J. Bot, 44, 371–379.

[51] Lakshmi, V. (2025). A Review on K nearest Neighbour Classification Technique in ML. Int. J. of Scientific Research in Science, Engineering and Technology.

[52] Bhatnagar, V., Poonia, R.C. (2021). Comparative Study of Supervised ML Algorithms for Healthcare Dataset Using Orange. In: Intelligent Learning for Computer Vision, Springer.

[53] Walji, K., Erraissi, A., Zakrani, A., Banane, M. (2025). Comparative Performance Evaluation of ML Algorithms in Sentiment Analysis. In: ICAISE 2024, Springer.

[54] Kalcheva, N. A., Karova, M. N. (2022). A Comparison of ML Classification Algorithms for English Works and their Translations. 57th Int. Conf. on Information, Communication and Energy Systems and Technologies (ICEST), IEEE.

[55] Dwivedi, S., Arya, C. (2016). Automatic Text Classification in Information Retrieval: A Survey. ICTCS '16, ACM.

[56] Sen, P.C., Hajra, M., Ghosh, M. (2020). Supervised Classification Algorithms in ML: A Survey and Review. In: Emerging Technology in Modelling and Graphics, Springer.

Mobile Applications that Incorporate AI for Information Search and Recommendation: A Systematic Literature Review

Mijael R. Aliaga, Jhosep S. Llacctahuaman, Carla N. Esquivel, Nemias Saboya
Facultad de Ingenieria y Arquitectura, Universidad Peruana Unión, Lima, Perú

***Abstract*—The inclusion of artificial intelligence (AI) has become essential for mobile application development, allowing improved personalization and optimization of the user experience. Over the past decade, smart mobile devices have been observed to enhance user experience across a variety of needs. The main objective of this study is to evaluate AI-powered mobile applications that utilize intelligent search mechanisms and more accurate recommendations and analyze their impact on addressing these user needs. The databases used were Scopus, ScienceDirect, Web of Science, and EBSCO. A filtering process using Prisma and a document quality assessment process was performed to select the most relevant articles. The study posed four questions related to the topic. The results showed that mobile apps for search and recommendation are mainly based on hybrid approaches (collaborative and content-based filtering) and deep learning (autoencoders, LSTMs/transformers, and BERT-type semantic retrieval embeddings), complemented by classic techniques (matrix factorization, SVM, K-NN, and trees/boosting) and contextual personalization (location, time, activity). It was concluded that these AI additions benefited users and met their search and recommendation needs. Furthermore, these mechanisms are advancing by leaps and bounds, as we now talk about more precise search and recommendation with voice, images, and even video.**

***Keywords*—AI algorithms; recommendation algorithms; mobile applications; search algorithms; AI for information**

I. INTRODUCTION

Artificial intelligence (AI) has become an important component of mobile app development [1], allowing for the implementation of recommendation algorithms that personalize and optimize the user experience. These algorithms can analyze a large amount of data to provide accurate and appropriate suggestions that enhance user interaction with the application.

For example, work-related applications such as LinkedIn, Indeed and Glassdoor stand out for connecting candidates and employers through personalized recommendations.

In the educational sector, applications like Coursera, Duolingo and Khan Academy integrate intelligent systems to customize the teaching content for each student, and in navigation, Google Maps and Waze use AI techniques to optimize routes and provide real-time contextual information. [1].

The integration of artificial intelligence into mobile applications is particularly important for information searches in the public and private sectors. Artificial intelligence provides a personalized learning experience and adapts to the needs and pace of each user.

This study provides an overview of how modern technology, especially artificial intelligence, has the ability to guide and recommend, allowing sugggestions to be personalized based on user needs, facilitating not only searches but also the exchange of information and advice between users. This fosters growth and innovation in the technological field.

Applications that include artificial intelligence significantly differentiate themselves in operational efficiency and user experience by providing a more intuitive and personalized interaction [1], [2]. This is in contrast to traditional applications, which often offer a more general and less adaptive experience.

Among the benefits of using mobile applications with artificial intelligence is the personalization of content, which improves the ability to offer accurate recommendations based on user behavior [2], [3]. These benefits not only improve user satisfaction but also increase app retention.

The most important factors contributing to the adoption and success of AI applications are ease of use, adherence to recommendations, and the ability to adapt to changing user needs [2]. These factors are important to ensure that applications are not only adopted but also thrive in a competitive market.

The need to study this topic stems from the exponential rise of mobile technology and its impact on users. As more people rely on their mobile devices, it's important to understand how artificial intelligence can improve this experience and what challenges must be overcome for effective implementation of this technology.

The main objective of this study is to conduct a systematic review to identify and analyze the use of artificial intelligence in mobile applications, which involve information search and recommendation within the application. It also analyzes the different AI algorithms and techniques that are commonly used, and determines the sectors in which it has been most frequently applied.

This study will recommend solutions to improve the integration of artificial intelligence into search and recommendation. Using a PICOC approach, the impact of these applications on the efficiency and effectiveness of everyday use will be analyzed.

II. Gaps in the Literature

Despite advances in the integration of artificial intelligence (AI) into educational mobile applications, significant gaps in the existing literature have been identified. Of the 58 studies analyzed, only seven focus specifically on the education sector. These works address use cases such as learning personalization, interactive content delivery, and automated assessment. However, none focus on job opportunity recommendations for university students or graduates, highlighting a lack of research on the application of AI for career guidance in this demographic.

Most AI-powered educational applications focus on features such as content classification, interactive simulations, and recommendation systems based on collaborative filtering, recurrent neural networks (GRUs), and text mining. These technologies, while effective in personalizing learning, do not address the specific needs of students seeking guidance in their transition to the labor market in which they work [2], [3], [4], [5].

Also, the lack of studies that integrate AI in mobile applications for job recommendation and professional development in the university environment limits the understanding of how these technologies can support students in planning their careers [5], [6], [7]. In addition, there is a lack of research that evaluates the effectiveness of these applications in real contexts, considering factors such as adaptability to different academic disciplines and the diversity of student profiles [7], [8], [9].

Therefore, this gap in the literature highlights the need for future research exploring the design and implementation of AI-powered mobile applications aimed at recommending job opportunities for university students and graduates. Such studies should consider the integration of technologies such as natural language processing, machine learning, and adaptive recommendation systems that can offer personalized guidance based on users' academic profiles and career aspirations [9].

III. Research Methodology

This study is based on a systematic review of the extraction of scientific articles from various scientific databases using the PRISMA methodology.

The PARSIFAL tool, a specialized platform for conducting systematic literature reviews, was used. This tool facilitates the organization and management of research through structured searches in academic databases, ensuring transparency and reproducibility of the process.

The research focuses on scientific publications related to artificial intelligence in mobile applications, given that the information is relevant and can be used. This study provides context for mobile applications on various platforms, offering a comprehensive view of the differences between applications that employ artificial intelligence and those that do not.

A. Research Questions

To assess the scope of mobile applications incorporating artificial intelligence (AI), particularly those dealing with user experience, application usability effectiveness and technological adoption, it is essential to have previous research demonstrating the results achieved. To achieve the objectives of this systematic literature review, the process has been structured into three main phases: review planning, review execution and review documentation, as shown in Fig. 1. This approach ensures a comprehensive and systematic analysis of mobile applications that integrate AI, comparing them with those that do not use this technology [5].

In addition, a detailed protocol was developed for the review, which established search filters and defined the study area in the context of mobile applications. The research questions guiding this review are shown in Table I.

TABLE I. Defining the Research Questions

Nro	Research questions	Reason
RQ1	What artificial intelligence methods, techniques, and algorithms are used in mobile applications for information recommendation and search?	Helps identify which AI methods and algorithms are used to improve recommendations and search in mobile apps.
RQ2	Which artificial intelligence algorithms or techniques were most frequently used in mobile information search and recommendation applications?	It allows you to know which algorithms are the most common and effective in mobile recommendation and search applications.
RQ3	In which areas or sectors were smart applications for information recommendation and search most widely used?	Helps you discover the sectors where AI-powered apps for recommendation and search have the greatest impact.
RQ4	What are the limitations and challenges in using artificial intelligence in mobile applications for information search and recommendation?	Help identify challenges and obstacles in using AI in mobile apps to improve search and recommendation.

B. Search Strategy

A systematic search was carried out in four databases: Scopus, ScienceDirect, EBSCO Host, and Web of Science (WoS) to compile research on the use of mobile applications that integrate artificial intelligence techniques. The time range considered was 2020 to 2026, allowing us to focus on recent and relevant studies.

Search strings were constructed using Boolean operators (AND, OR, NOT), combining terms related to artificial intelligence, mobile applications, search and recommendation functionalities, as well as user experience and technical aspects. In addition, exclusion filters were applied in some databases to exclude publications related to the medical field. The exact strings used are presented in Table II.

C. PRISMA Method

When conducting a systematic review on artificial intelligence in mobile applications, PRISMA helps structure the review process to be clear and reproducible. This is especially important in a field as dynamic as AI, where the literature can be extensive and varied [10].

TABLE II. SEARCH CHAIN FOR EACH DATABASE

Database	Search string
Web of Sciencie	("artificial intelligence" OR "ai" OR "machine learning" OR "deep learning" OR "neural networks") AND ("mobile application" OR "mobile app" OR "smartphone application" OR "app") AND ("search" OR "recommendation" OR "suggestion" OR "discovery" OR "ranking function" OR "search algorithm") AND ("user experience" OR "ui" OR "ux" OR "personalization" OR "interaction") AND ("algorithm" OR "model" OR "system" OR "framework")
Science Direct	("artificial intelligence" AND "mobile applications")AND ("machine learning" OR "deep learning") NOT ("health" OR "covid 19" OR "Medical")
Scopus	("artificial intelligence" OR "ai" OR "machine learning" OR "deep learning" OR "neural networks") AND ("mobile application" OR "mobile app" OR "smartphone application" OR "app") AND ("search" OR "recommendation" OR "suggestion" OR "discovery" OR "ranking function" OR "search algorithm") AND ("user experience" OR "ux" OR "personalization" OR "interaction") AND ("algorithm" OR "model" OR "system" OR "framework")
Ebsco	("Artificial Intelligence" OR "IA" OR "Machine learning" OR "deep learning") AND ("mobile applications" OR "Mobile apps") AND ("algorithm") NOT ("Health" OR "Covid 19" OR "Medical")

In this review, we begin by identifying a large number of papers related to AI in mobile applications. Using the PRISMA flowchart, we document each stage of the selection process, from the initial identification of 6,138 articles to the final selection of 58 relevant articles. This flowchart facilitates understanding and ensures transparency of the selection process, as shown in Fig. 1.

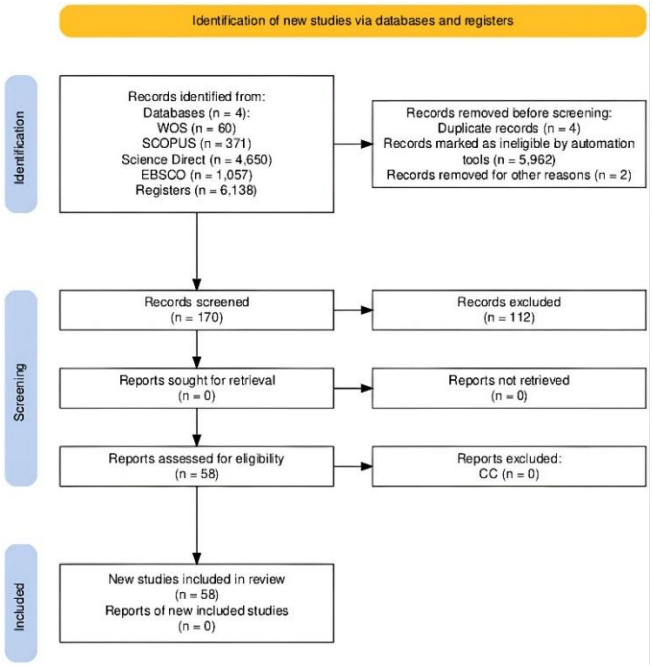


Fig. 1. Article selection process.

In accordance with PRISMA, we pre-specified and fully reported the eligibility criteria used to screen the studies. We included peer-reviewed articles on mobile apps that implement AI techniques and that examine user outcomes, such as acceptance, behavior, or satisfaction with everyday use. We excluded records that did not focus on mobile apps, that did not directly employ AI, or that were non-empirical works, as well as duplicates.

Study selection followed a two-stage process documented in a PRISMA flowchart with the counts of records identified, reviewed, excluded, and included, along with the reasons for exclusion. For the studies, we extracted key variables and conducted a narrative synthesis. In the context of AI for mobile apps, this synthesis highlights how search and recommendation algorithms relate to user satisfaction and acceptance and identifies areas where evidence remains limited.

D. Stages

Moving on to the stages, this process of selecting and filtering papers for this study is carried out through a systematic approach and in successive stages, applying specific search, filtering, inclusion, and quality criteria in the selected databases (WoS, Science Direct, Scopus, and EBSCO). The initial search generated a high volume of papers, which were reduced and evaluated through six main stages, detailed below:

1) Stage 1 – No filters: This is the initial phase of searching for all possible information related to the research topic without using any exclusion or selection criteria. The main objective here is to thoroughly review and gather the most appropriate data, documents, or articles, although some may not be useful in the end. The search strings used in this stage are shown in Table II.

2) Stage 2 – Basic filters: Internal filters were applied to each database to restrict results to the last five years (2020 to 2026), select open access scholarly and review articles, and limit the languages to English and Spanish for WoS, Science Direct, and Scopus, and English only for EBSCO HOST. After this step, 1,485 records remained that met these requirements.

3) Stage 3 – Initial inclusion and exclusion criteria by database: This consisted of applying our specific inclusion and exclusion criteria mentioned in the methodology, which varied slightly depending on the database. Articles with sources that were not part of the study were discarded, such as: “Rehabilitation”, “Geochemistry”, “Operations Research”, “Management Science”, “Physics or Mechanics”, “clinical”, “health”, “Covid”, “Human”, “Healthcare”, “Autism”, “covid-19”, “Diabetes”, “Diagnosis” and “impaired”, which led to obtaining 176 articles.

4) Stage 4 – Duplicate removal: Duplicate articles across different databases were identified and removed. This is critical to ensuring the quality and accuracy of the analysis, yielding 170 independent references.

5) Stage 5 – Specific inclusion and exclusion criteria: Additional inclusion criteria were applied, such as analyzing the impact of AI on mobile applications and recommendation algorithms, and exclusion criteria were applied, eliminating

studies outside the fields of engineering, AI, and mobile applications. The inclusion and exclusion criteria used are shown in Table III and Table IV.

TABLE III. INCLUSION CRITERIA

Nro	Inclusion Criteriaa	Reason
CI1	Articles that analyze the impact of AI on users' daily lives and the technology's acceptance in mobile applications.	This criterion is important for understanding the impact of this technology on the daily lives of users who adopt it. It also allows us to assess the technology's viability in the market and its growth potential.
CI2	Articles that address Mobile Applications.	This is important because it ensures that the selected articles provide information directly applicable to the development and use of AI in the mobile environment.
CI3	Articles that address artificial intelligence (AI) applied to mobile application development.	These studies are key to understanding the methodologies, tools, and architectures used to integrate AI into these types of applications.
CI4	Articles that discuss machine learning and digital technologies that contribute to improving user efficiency and convenience.	Including studies that analyze these technologies allows us to understand how machine learning models optimize the functionality of mobile applications, benefiting personalization, speed and precision in interaction.
CI5	Articles that include recommendation algorithms in mobile applications.	This ensures that research focusing on the development and effectiveness of recommendation systems is investigated.

TABLE IV. EXCLUSION CRITERIA

Nro	Exclusion Criteria	Reason
CE1	Duplicate articles	Removing duplicate articles is standard in review research; this prevents repetition and makes sure each study offers fresh, relevant information.
CE2	Studies not related to engineering, software development, mobile applications, computer science, artificial intelligence, or digital technologies.	Focusing on these areas ensures the reviewed studies are both technically sound and practically useful, excluding research that doesn't offer applicable knowledge for developing or implementing AI in mobile apps.
CE3	Research into digital technologies for non-mobile platforms, such as desktop applications or back-end systems without mobile functionality.	By focusing only on studies of mobile platforms, we can be sure the articles we choose offer insights specific to the unique challenges and benefits of mobile environments.
CE4	Research that addresses AI technologies, machine learning, or algorithms, but is not applied in the context of mobile applications	This selection process eliminates studies unrelated to mobile apps, concentrating the review on research relevant to the unique challenges and environment of mobile devices.
CI5	Posts focused on AI applications that are not related to mobile apps or recommendation algorithms.	Ensures that selected articles focus on specific application areas that seek to improve the user experience on mobile devices.

6) Stage 6 – Quality assessment: Articles that passed the preceding stages were evaluated using an eight-question quality checklist. A minimum score of 4 was required to proceed. The criteria assessed methodological rigor and the extent to which user acceptance of AI in mobile applications was analyzed. Table VI lists the 58 studies that met the threshold and advanced to the final stage.

The general summary of the total number of articles we obtained at each stage can be seen in Table V.

Finally, we evaluated the relevance of each article to our quality questions, awarding 1 point for a complete answer, 0.5 for a partial answer, and 0 for no direct evidence. Of the 58 articles evaluated, all achieved an average of 5.45 points, confirming their suitability for in-depth analysis. Each article's results are shown in Table VI.

Articles that passed Stage 6 were evaluated in a final phase to determine their contribution to the research questions. This evaluation ensured that the studies aligned with the established thematic lines: efficiency improvement, popularity, technological trends, use cases, impact of recommendation algorithms, and challenges in user experience. Specific research questions addressed the following topics:

- Improving efficiency through AI-based mobile applications.
- Current technological trends in AI applications.
- AI use cases in the fields of employment and education.
- Impact of recommendation algorithms on user experience.

E. Keyword Analysis

To identify the thematic axes, a co-occurrence map was created with VOSviewer 1.6.x, applying the association strength algorithm of van Eck and Waltman [11]. We worked exclusively with Scopus, the most cited database and with the greatest initial coverage in this study. Each node of the graph represents an author's keyword; its size indicates the frequency, and the thickness of the links reflects the intensity of co-occurrence.

- Fig. 2: We started with a corpus of 371 articles. The 170 terms we chose fell into five groups, with machine learning, artificial intelligence, and mobile apps being the most common. This shows how important AI is for mobile devices.
- Fig. 3: Cleaned-up corpus (31 articles). Following PRISMA filtering and quality control, the resulting graph showed 16 terms in two clusters:
    - Technical (red): machine learning, deep learning, natural language processing, mobile applications.
    - Evaluation with users (green): human, controlled study, procedures.

The reduction from five to two clusters indicates that the most rigorous evidence focuses on: i) AI techniques for mobile personalization andii) experimental studies with users. Furthermore, the absence of terms such as "job

recommendation" or "career guidance" corroborates the previously identified gap: the lack of research that provides professional guidance to students and graduates through AI-powered mobile applications.

TABLE V. DATABASE RESULTS

Database	Results						
	Stages						
	Stage 1	Stage 2	Stage 3	Stage 4		Stage 5	Stage 6
WOS	60	36	23		23	10	58
SCOPUS	371	78	46	5	41	31	
Science Direct	4650	1127	69		69	7	
Ebsco Hostt	1057	244	38	1	37	10	
Total	6138	1485	176	-6	170	58	58

TABLE VI. MOBILE APPS WITH AI: METHODS AND TECHNOLOGIES

Method / Technique	Recommendation Algorithm	What is it for?
Filter and Recommendation [12], [13], [14] Systems Evaluation and QA [15], [16] Data Extraction and Preprocessing [12], [17] Statistical Analysis [18], [19] Sequential Modeling and Transitions [19], [20] Probabilistic Methods and Game Theory [21] Classical ML Algorithms [20], [22], [23], [24] Hyperparameter Optimization [20], [25] Deep Learning and Neural Networks [23], [24], [25], [26], [27], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37], [38], [39] Architectures Advanced Neural Processing [29], [31], [40] Image and Sensor Processing [14], [31] Natural Language and Text Processing [35], [41], [42], [43] Dialogue and Text Generation [41], [44] User Profiling and Behavior [15], [45] Ranking and Output Combination [19], [20], [28], [33], [34], [38], [42], [44]	Machine Learning (generic) [14], [23], [28], [30], [31], [33], [34], [38] Tree and Ensemble Algorithms Random Forest [22], [24] Random Survival Forest [23] Logistic Regression [17], [19] Gaussian Process Regression [21] Clustering and K Means Neighbors [15] K-Nearest Neighbors (KNN): [23], [40] Natural Language Processing (NLP) [16] Neural Networks (Deep Learning) CNN [20], [27], [29], [32], [33], [34], [35], [36], [37], [38], [41] RNN [44] LSTM [36], [37], [44] GRU ANN SNN (also: Generic Deep Learning) Supervised Learning Algorithm (Generic) Combined / Advanced Vision, Methods: Mask R CNN, Faster R CNN, K Means Algorithm	Content and feature based recommendation [15], [28], [32], [37], [38], [42], [43], [45] Behavior-based and collaborative recommendation [44] Prediction based on biomedical and clinical data [18], [22], [23], [27], [34], [35] Detection based on security patterns and anomalies [19], [30] Optimization based on performance and resources [20], [21], [24], [25], [33], [36], [40] Adaptive interfaces based on user signals [14], [16], [17], [26], [29], [31], [39], [41]
Method / Technique	**Search Algorithm**	**What is it for?**
Machine Learning [46] Deep Learning [47] Random Forest [48] Neural Networks (CNN) [47] Data Preprocessing, [48] KNN SVM	NLP [46] CNN [47] Logistic Regression [48]	Search based on efficiency and user experience [46] Classification based on visual features and deep learning[47] Analysis based on text mining [48]

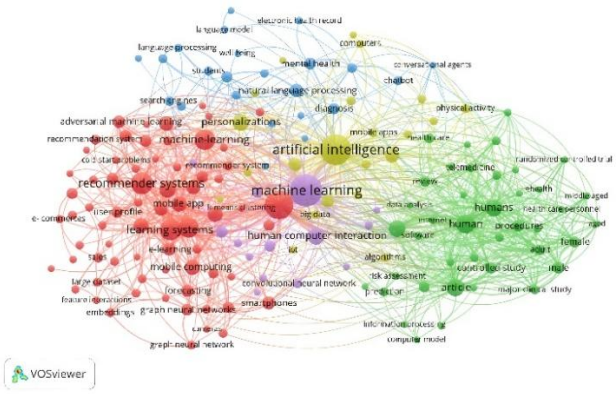


Fig. 2. Image of Scopus keywords before filters.

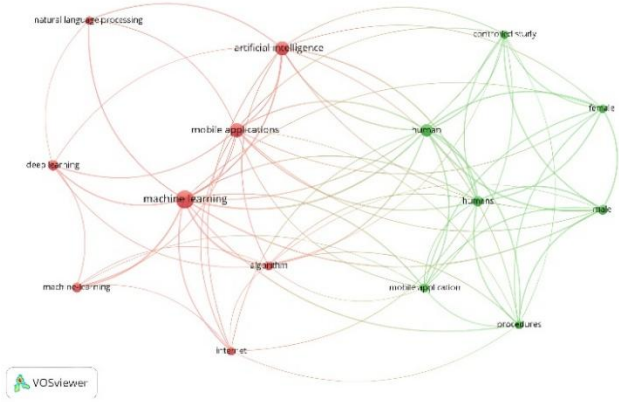


Fig. 3. Scopus keywords image after applying the filters.

IV. RESULTS

Below, the results obtained from the Parsifal platform will be explained, both as selected and accepted articles, and the articles that respond to the research questions posed in Table I will also be detailed, each question with its respective answer.

A. Descriptive Results

1) Article by source: The pie chart (see Fig. 4) summarizes the relative contribution of each database to your search:

- Science Direct 39.7 % provides the majority of the results, almost four out of ten.
- Scopus 25.9 % is the second most prominent source, with just over a quarter.
- EBSCO remains the most important at 21.3 %, covering approximately one-fifth of records.
- Web of Science 13.2 % contributes the smallest portion, about one in eight results.

These four databases cover the entire range of findings, with Science Direct leading by far and Web of Science having the smallest share.

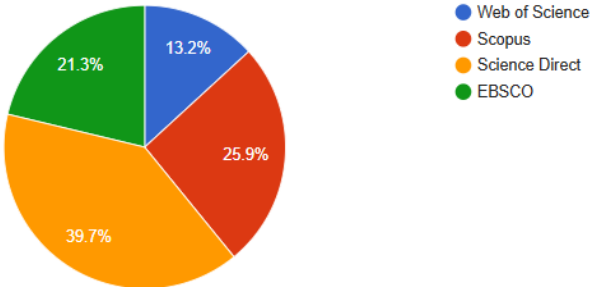


Fig. 4. Percentage distribution of results by database.

2) Accepted articles: For this research, based on the databases and search strings used in each, a total of 6,138 articles were selected, and after a multistage filtering process, only 58 were ultimately accepted for the study (see Fig. 5). Web of Science shows the highest acceptance rate (16.7%), followed by Scopus (9.4%). EBSCO has a medium-low rate (1.1%), while Science Direct, despite being the source with the most articles retrieved, provides the lowest proportion of accepted articles (0.2%).

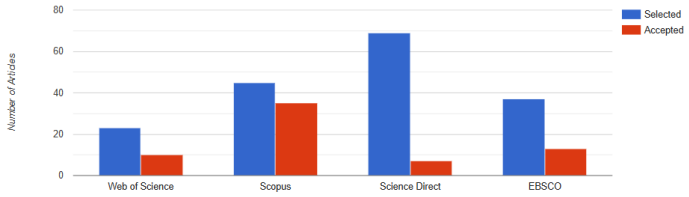


Fig. 5. Number of articles selected and accepted by the database.

B. Answer to Each Research Question

The answers to each of the research questions are presented below. Each answer will be based on the content of the selected articles that provide specific evidence.

1) What artificial intelligence methods, techniques, and algorithms are used in mobile applications for information recommendation and search?

Mobile applications that integrate artificial intelligence for information recommendation and search employ a variety of advanced methods and techniques, including machine learning algorithms (such as decision trees, logistic regression, Random Forest, KNN, and clustering), deep learning techniques (including convolutional neural networks –CNN–, recurrent networks –RNN, LSTM, GRU–, and advanced architectures such as Mask R-CNN and Faster R-CNN), as well as hybrid approaches that combine recommendation, classification, and ranking of results. Other methods, such as data extraction and preprocessing, feature selection, and natural language processing (NLP), are also used to analyze user behavior and to improve the understanding and retrieval of textual information.

2) What AI algorithms or techniques were frequently used in mobile information recommendation applications?

The reviewed studies emphasize the use of neural networks (including various deep learning models) because they simultaneously learn item characteristics and user preferences, generating highly accurate recommendations even with heterogeneous data (text, images, clicks). Traditional machine learning algorithms, such as SVM, KNN, or trees, are also used when a lightweight model that can run quickly on the device is needed. Many studies combine these two types with clustering, which groups users or content to initiate recommendations when there is little historical information. Content filtering has proven useful for suggesting items similar to those already viewed, while natural language processing (NLP) techniques help interpret queries or reviews and enrich the user profile. Finally, methods such as Random Forest occasionally appear for classification tasks within the system, but are not the core of the recommendation process, as summarized in Table VII.

TABLE VII. MOST FREQUENTLY USED ALGORITHMS

Articles	Tendencies
[14], [15], [16], [30]	Filtering and Recommendation (by content)
[24], [25], [28], [45]	Deep learning
[14], [23], [28], [30], [31], [33], [34], [38]	Machine learning
[16], [46]	Natural language processing
[20], [27], [29], [32], [33], [34], [35], [36], [37], [38], [41]	Neural Networks
[48]	Random Forest
[19], [20], [28], [33], [34], [38], [42], [44]	Clusters

3) In which areas or sectors were intelligent applications for information recommendation and search most used?

The sectors where intelligent applications for information recommendation and search were most widely applied were healthcare, technology, entertainment, and education. The healthcare sector stood out for its personalized diagnostic and monitoring systems; technology and entertainment demonstrated advanced use of algorithms, such as collaborative

filtering, NLP, and neural networks to personalize content, with high potential for adaptation to the work context. Although no applications aimed at job recommendation were found in education, similar techniques were used, demonstrating the transversality of these technologies in different contexts, as shown in Table VIII.

TABLE VIII. SMART APPLICATIONS IN KEY SECTORS FOR INFORMATION RECOMMENDATION AND SEARCH

Sector / Area	Use Cases	Application Type	Technology / AI / Algorithms
Health [18], [22], [23], [25], [26], [27], [29], [39], [44], [46], [49], [50], [51], [52], [53], [54], [55], [56], [57], [58], [59], [60], [61]	1. Personalized and Predictive Clinical Care 2. Support and Automation for Professionals 3. Advanced Analytics and Recommendations	Optimizing medical processes and clinical support with smart technologies. Advanced diagnostics, treatment customization, and intelligent monitoring.	1. Machine Learning and Deep Learning 2. Clinical and Biomedical Processing 3. Multimodal Analysis and Media 4. Adaptive Systems and Personalized Recommendations 6. Technological Infrastructure and IoT
Digital entertainment [38], [62]	1. Movie recommendations on platforms like Netflix, Amazon Prime, and Disney. 2. Personalized short video recommendations on short-form video platforms.	Intelligent recommendation systems for digital content (movies and short videos).	1. Collaborative and Content-Based Filtering Techniques - Knowledge-Based Systems and Hybrid Approaches 2. Deep Learning and Stream Processing 3. User Interest Modeling
Education [15], [30], [40], [47], [63], [64], [65]	- Personalization and learning recommendations - Interactive content delivery - Automated content management, classification, and evaluation	Smart educational apps for: - Personalized recommendations - Automated assessment - Interactive simulations - Content classification - Learning personalization	1. Recommender systems with collaborative filtering 2. GRUs (Recurrent Networks) for adaptive recommendation 3. Text mining and SNNs 4. AI with pattern recognition 5. Physical and 3D simulations 6. NLP and machine learning 7. Knowledge tracking with convolutional CKT
Agriculture [66], [67], [68]	Animal health diagnostics, crop disease detection, and IoT monitoring.	Mobile and smart apps for animal diagnostics, agricultural monitoring, and personalized recommendations.	1. Data-driven diagnosis 2. TAM (Technology Acceptance Model) 3. Convolutional Neural Networks (CNN) 4. Inception v3 5. Deep Metric Learning
Technology [16], [28], [69], [70], [71], [72]	Content personalization, user satisfaction prediction, semantic retrieval, interface and news personalization.	Intelligent apps and platforms for behavioral analysis, contextual personalization, and adaptive interaction.	1. VADER 2. Machine learning 3. Natural language processing (NLP) 4. Matrix factorization (MF) 5. GNNs + HAMP 6. 1. FIS (Feature Interaction Search) 7. 3. Recommendation algorithms
Finance [17]	Smart investment advice through mobile apps.	Multidimensional data analysis to recommend appropriate options.	1. NLP 2. Question and Answer (Q&A) Systems
Cybersecurity [43], [45]	1. Malware detection in Android through API call sequence analysis. 2. Botnet detection in Android through static app analysis.	1. Optimized algorithms to reduce redundant exploration paths. 2. Intelligent mobile threat detection framework.	1. Machine Learning Classifiers 2. Pruning Search 3. Graph Transformation 4. Permissions Review, APIs 5. Supervised ML
Technological Infrastructure [19]	Predicting non-functional attributes (QoS) in cloud services.	Intelligent time-sensitive models for quality prediction.	1. Sequential Modeling (GRU) 2. Probabilistic Discrimination (GAN)
Trade [14], [73]	Voice shopping, personalized recommendations on smart kiosks.	Voice assistants and self-service kiosks with product customization.	1. Voice Assistants (VAs) 2. LLMs 3. Deep learning 4.RFID
Transportation / Automation [36], [41]	1. Emotion recognition and assistance in automated driving. 2. Analysis of user feedback on electric vehicle charging stations.	1. Mobile apps with adaptive neural networks based on device context. 2. Text classification and sentiment analysis.	1. Deep Learning (DNN) 2. AdaScale for real-time dynamic scaling. 3. Deep Learning 4. Transformers (BERT, XLNet) 5. CNNs, LSTMs
Computer Vision / AI [74], [75]	Object recognition in images/videos and analysis of ceramics in cultural heritage.	Apps with neural networks that detect, classify, and group visual objects on the device.	1. ML on mobile devices 2. Faster R-CNN, Mask R-CNN 3. K-means
Mobile Sensing / IoT / Privacy [42]	Crowdsensing of spatial data (location, audio, etc.) for tasks such as environmental monitoring.	Mobile apps with incentives to collect data while protecting privacy.	1. Stackelberg Mechanism 2. Privacy-aware data sharing
Telecommunications / Energy [21]	User opinion analysis using text and sentiment mining.	Mobile applications that extract knowledge from textual feedback.	1. Text mining 2. NLP for topics and emotions
Energy / Smart Home [33], [48]	Energy optimization in homes and buildings through occupancy detection.	Systems with IoT sensors and predictive models to control HVAC.	1. Digital Twins 2. ML for Occupancy 3. CNNs + Wavelet Transform

Tourism / Personal Mobility [37]	Recommendation of places to visit based on user behavior.	Recommendation systems based on location and check-in sequence.	1. InfAM 2. Embeddings 3. Residual network

4) What are the limitations and challenges of using artificial intelligence in mobile applications for information search and recommendation?

Limitations and challenges in the use of artificial intelligence in mobile applications for information search and recommendation were identified in the quality and availability of data, effective personalization in diverse contexts, and ethical concerns related to user privacy. In sectors such as healthcare and education, implementation is limited by the sensitivity of the information and the need to comply with strict data protection regulations. Although technologies such as neural network filtering have proven effective, their performance can always be affected by incomplete or biased data. Poor interoperability between platforms and a lack of common standards make it difficult to integrate these systems into large-scale mobile applications. These aspects are summarized in Table IX.

TABLE IX. LIMITATIONS AND CHALLENGES OF USING INTELLIGENCE IN MOBILE APPLICATIONS FOR INFORMATION SEARCH AND RECOMMENDATION

Search Algorithm	**Limitations**	**Challenges**
- KESS (Keyword Growing + Systematic Search) - Content-based Image Retrieval + Deep Metric Learning (Contrastive Loss)	Search results took 43 seconds longer on average. [46] Inaccurate dance pose evaluation. Limited accuracy in real-life scenarios (71%), biased recognition of foods from one's own country. [73] QoS data shortage; cold start with new users/services.[40] Variable quality and possible noise in crowdsourced data; reliance on student data.[60] Missing important information and challenging viewpoints due to algorithms.[15] Limited accuracy in some visual categories. [22], [53] Validation results in animal models (rats) do not fully reflect human effects. [66]	Ensure user trust and data quality. [27] Improve the accuracy of object recognition and visual recommendations. [39] Various hardware conditions and limited resources on mobile devices. [25] Accurate QoS prediction, improving user experience. [40] Reduce computational cost without losing accuracy. [65] Capture complex user behavior patterns. [37] Improve infrastructure and data quality, ensuring effective personalization. [75] Improve the accuracy of predicting dynamic user interests. [39]
Recommendation Algorithm	**Limitations**	**Challenges**
- Collaborative Filtering - Hybrid Filtering - GRU–GAN - Stacked Autoencoders - Graph Autoencoders - Neural Networks (Deep Learning, Transfer Learning) - Linear and Lasso Regression Models with Genomic Data - Collaborative Filtering (CF) and Matrix Factorization (MF) - Collaborative Filtering, Stacked Autoencoders, Graph Autoencoders - Recommendation Algorithms Based on News Personalization - Adaptive and Personalized ML Techniques - Recommendation Algorithms Based on Medical Evidence + Generative AI (ChatGPT-4) - Diagnostic and Recommendation Algorithms - Conscious Influence in Successive POI Recommendations - CAT-MF Rec Based on DIEN and Cross-Attention Mechanism	- Inaccurate or misguided recommendations (investors, travel itineraries, clinical treatment). [6], [40], [46], [50] - Difficulty distinguishing explicit vs. implicit feedback. [38] - Latency and data quality limit dynamic user profile updates. [18], [27] - Anthropomorphic interfaces or AI do not always increase trust. [63] - Dependencia ciega del usuario sin entender cómo funciona el algoritmo. [15], [37] - Falta de consenso en autoevaluación del estado de ánimo y otros autoinformes. [54] - Modelos grandes (LLM, DL) difíciles de ejecutar en móviles. [65] - Necesidad de adaptación constante a condiciones del hardware (batería, CPU, red). [25] - Complejidad computacional y procesamiento multimodal (rostro, voz, biometría) elevan el consumo de recursos. [26] - Preocupaciones por privacidad y almacenamiento de datos sensibles. [29], [39], [54], [56], [57] - Algoritmos generativos aún con limitaciones de precisión y seguridad. [57] - Alto tiempo de implementación y escasa integración con sistemas existentes (historial electrónico, plataformas de terceros). [56]	- Integrate contextual cues (location, time, usage, history) [86] - Combine demographic, genomic, or microbiome data to personalize clinical interventions [37], [52] - Incorporate continuous - Fusionar filtrado colaborativo, basado en contenido y demográfico sin comprometer escalabilidad. [53] - Integrar modelos de lenguaje grande (LLM) para más precisión y velocidad. [38], [39], [46], [73] - Adaptar modelos a dispositivos con recursos limitados y validar su desempeño. [25], [72] - Integrar reconocimiento de voz, rostro, emociones y texto (PLN) en recomendaciones. [51] - Fusión eficiente de múltiples modalidades para mejorar la personalización. [27], [39], [45], [46], [57] - Ajustar recomendaciones a expectativas, progreso y "olvido" del usuario en tiempo real. [27], [39], [42], [46], [57] - Identificar factores clave que influyen en preferencias cambiantes. [18] - Diseñar UI accesibles que permitan calibrar el nivel de confianza del usuario. [63] - Mejorar la transparencia para evitar cámaras de eco y fomentar la comprensión del sistema.[15], [23] - Garantizar robustez en entornos diversos y culturas distintas. [75] - Aumentar la empatía de la IA para diferentes perfiles y contextos de compra o uso. [41] Salud: personalizar rutas de aprendizaje/coach, recomendaciones de medicación seguras y evaluación de resultados clínicos. Educación: generar rutas de aprendizaje personalizadas y adaptativas. Entretenimiento & otros: integrar contenidos y experiencias altamente personalizadas.[7], [46]

V. DISCUSSION

In this systematic review, we examine how artificial intelligence is integrated into mobile applications for information search and recommendation, addressing four research questions. We reduced the initial corpus from 6,138 records to 58 included studies after applying prespecified eligibility criteria—publication window (last five years), English/Spanish language, software-engineering relevance, and an explicit focus on search or recommendation algorithms. Among the included studies, 39.7% were retrieved from ScienceDirect, 25.9% from Scopus, 21.3% from EBSCOhost, and 13.2% from Web of Science, reflecting differences in coverage and accessibility across databases.

Regarding methods and algorithms (RQ1), the articles studied confirm the dominance of CNN neural networks for visual content and RNN/GRU/LSTM for interaction sequences combined with classic machine learning techniques (decision trees, SVM, k-NN, Random Forest). This coexistence responds to a compromise between precision (achieved by deep models) and lightness (necessary for on-device processing). In addition, there is a growing use of hybrid approaches that combine collaborative filtering and classification, as well as PLN can also be used to obtain knowledge and information from text corpora. Taken together, these techniques show the potential of machine learning algorithms to reduce the workload of human analysts on specific tasks [76].

Regarding the technological trends found in the study (RQ2), the analysis shows that neural networks appear in more than half of the studies, followed by classic collaborative filtering, content-based filtering, and clustering. This pattern indicates that developers prioritize models capable of learning deep representations when resources are available, while using lightweight collaborative, content-based, and clustering techniques for auxiliary tasks such as pre-filtering, cold start, or segmentation. The still-spot presence of Random Forest and other ensembles provides a niche for interpretable and efficient algorithms, especially in domains where system explanation is critical. Computer tools that make users think about products, services or content according to their interests, characteristics and behavior using algorithms [77].

Sectors that were applied (RQ3). Four sectors concentrate the majority of the implementations:

Healthcare. Personalized diagnostic and monitoring systems predominate, requiring clinical precision and regulatory compliance.

Technology and entertainment. Collaborative filtering, NLP, and neural networks are used to personalize content in real time.

Education. Resource recommendation systems, automated assessment, and interactive simulations are being used, although research on job or internship recommendations for students is lacking.

General services (travel, finance, commerce). They use AI to suggest routes, products, or investments, but face greater challenges in terms of trust and regulation.

The underrepresentation of the education sector in job search tasks constitutes a clear opportunity for research and transfer.

Limitations found (RQ4). The identified limitations converge on three axes: Data and accuracy. Scarcity or noise in training sets, cultural biases, and latency when updating profiles affect the quality of recommendations. Resource constraints. Large models increase CPU, battery, and bandwidth consumption; mid-range or low-end devices affect performance. Trust, privacy, and integration. Users often rely on recommendations they don't understand, while concerns persist about the use of biometric data and interoperability with external systems. These limitations pose immediate challenges: optimizing models for edge computing without sacrificing accuracy; efficiently fusing contextual and multimodal signals; designing explainable interfaces that enhance algorithmic literacy; and aligning solutions with ethical and regulatory frameworks, especially in healthcare and finance.

Strengths and limitations: The use of the PRISMA methodology and the coverage of four recognized databases increase the comprehensiveness and transparency of the review.

This systematic review shows that AI drives personalization and efficiencies in mobile search and recommendation applications. Despite this, its widespread adoption faces challenges related to data, technical resources, and user trust. Mapping the different algorithms, sectors, limitations, and opportunities identified a clear gap in the educational field. Very few studies explore the use of mobile AI to facilitate access to academic or training information. Therefore, in future work, it allows us to draw a broader research focused on the development of search and recommendation algorithms that not only prioritize efficiency and precision, but also guarantee transparency in their processes and the protection of sensitive user data. This involves exploring explainability techniques in AI that allow understanding how recommendations are generated and avoiding biases in the information offered. It is also necessary to investigate how these applications can be effectively integrated into learning environments, encouraging the personalization of educational content and adaptability to different learning styles.

Finally, the future of this line of research should focus on the creation of more useful, responsible and ethical mobile applications that respond to the needs of the educational sector, promote pedagogical innovation and contribute to strengthening the quality of the teaching-learning process.

VI. Conclusion

This systematic review study explored how AI is applied in various mobile applications for information recommendation and search, and addressed methods, techniques, and algorithms used in this field, as detailed in the research questions above.

Mobile applications integrated with: AI methods, techniques and algorithms used:

It was observed that these mobile applications employ machine learning algorithms (collaborative filtering, regression, and decision trees) and deep learning algorithms (especially neural networks and their derivatives). It was also observed that these hybrid approaches, which combine recommendation and classification techniques, are used. These algorithms are designed to adapt to the processing limitations of mobile devices, thus improving the accuracy and personalization of search and recommendation results.

Most frequent algorithms in the study:

According to research, the algorithms most commonly used in mobile applications are collaborative filtering, deep neural networks, and hybrid techniques. These approaches are the most widely applied due to their effectiveness in improving the accuracy of results and personalizing the user experience. This trend highlights the importance of these algorithms in improving user interaction with mobile search and recommendation applications.

Different sectors that cover search or recommendation algorithms:

A wide variety of algorithms was observed being applied across different sectors. The healthcare and entertainment sectors were found to be leading the use of search and recommendation algorithms. These sectors have adopted advanced technologies to improve the user experience. It was found that the implementation of recommendation algorithms is significantly lower in the education sector, suggesting an opportunity for improvement and expansion in this field.

Different limitations and challenges in applications for search and recommendation:

After observing mobile applications, it is clear that each one has its own recipe for searching and recommending content. Some rely on well-established techniques, such as collaborative filtering or classic KNN. Others, however, rely on the latest: deep neural networks, context-aware models, and even generative algorithms. The limitations detected include low accuracy, scarcity of interaction data, high dependence on context or user participation, and computational limitations on mobile devices. Recurring themes among the challenges are identified, such as effective personalization, ethical and secure integration, user trust, data quality, and adaptability to diverse and changing contexts. Looking ahead, the development of these applications should focus on making algorithms more accurate, models more transparent, and human interaction more fluid, all without compromising privacy or system performance.

REFERENCES

[1] C.-P. Julio Gustavo, P.-V. Amanda Beatriz, L.-C. Edison José, and G.-F. Segundo Alfredo, "The influence of artificial intelligence on the development of university education," Revista Científica Arbitrada de Investigación en Comunicación, Marketing y Empresa REICOMUNICAR, vol. 6, 2023, doi: https://doi.org/10.46296/rc.v6i12edespoct.0171.

[2] M. Mercedes et al., "Use of artificial intelligence in mobile devices," vol. 6, 2022, doi: 10.47230/unesum-ciencias.v6.n3.2022.460.

[3] A. Kaushik, "A Smart Learning Environment Enabled by AI- Powered Recommendations," 2025. Accessed: Aug. 24, 2025. [Online]. Available: https://ijrpr.com/uploads/V6ISSUE4/IJRPR43365.pdf

[4] Y. Li et al., "An empirical study of AI techniques in mobile applications," Journal of Systems and Software, vol. 219, p. 112233, Jan. 2025, doi: 10.1016/j.jss.2024.112233.

[5] M. Aucancela, A. G. Briones, and P. Chamoso, "Educational Recommender Systems: A Systematic Literature Review," in The Barcelona Conference on Education 2023: Official Conference Proceedings, The International Academic Forum(IAFOR), Nov. 2023, pp. 933–951. doi: 10.22492/issn.2435-9467.2023.74.

[6] A. M. Vieriu and G. Petrea, "The Impact of Artificial Intelligence (AI) on Students' Academic Development," Educ Sci (Basel), vol. 15, no. 3, Mar. 2025, doi: https://doi.org/10.3390/educsci15030343.

[7] P. Bahalkar, D. Y. Patil, V. P. Society, P. Peddi, and S. Jain, "AI-Driven Career Guidance System: A Predictive Model for Student Subject Recommendations Based on Academic Performance and Aspirations," 2024. Accessed: Aug. 24, 2025. [Online]. Available: https://healthinformaticsjournal.com/index.php/IJMI/article/view/781/738

[8] M. Y. Mustafa et al., "A systematic review of literature reviews on artificial intelligence in education (AIED): a roadmap to a future research agenda," Dec. 01, 2024, Springer. doi: 10.1186/s40561-024-00350-5.

[9] F. Trujillo, M. Pozo, and G. Suntaxi, "Artificial Intelligence in Education: A Systematic Literature Review of Machine Learning Approaches in Student Career Prediction," J Technol Sci Educ, vol. 15, no. 1, pp. 162–185, 2025, doi: https://doi.org/10.3926/jotse.3124.

[10] J. A. Hoyos Muñoz and L. J. Henao Tamayo, "Key Factors in Technology Adoption Processes," 360: Revista de Ciencias de la Gestión, no. 7, Nov. 2022, doi: 10.18800/360gestion.202207.009.

[11] A. Kirby, "Exploratory Bibliometrics: Using VOSviewer as a Preliminary Research Tool," Publications, vol. 11, no. 1, p. 10, Feb. 2023, doi: 10.3390/publications11010010.

[12] M. Lan and X. Zhou, "A qualitative systematic review on AI empowered self-regulated learning in higher education," NPJ Sci Learn, vol. 10, no. 1, Dec. 2025, doi: 10.1038/s41539-025-00319-0.

[13] B. Klimova and M. Pikhart, "Exploring the effects of artificial intelligence on student and academic well-being in higher education: a mini-review," 2025, Frontiers Media SA. doi: 10.3389/fpsyg.2025.1498132.

[14] E. Gabarron, D. Larbi, O. Rivera-Romero, and K. Denecke, "Human Factors in AI-Driven Digital Solutions for Increasing Physical Activity: Scoping Review," 2024, JMIR Publications Inc. doi: 10.2196/55964.

[15] Y. R. Du, "Personalization, Echo Chambers, News Literacy, and Algorithmic Literacy: A Qualitative Study of AI-Powered News App Users," J Broadcast Electron Media, vol. 67, no. 3, pp. 246–273, May 2023, doi: 10.1080/08838151.2023.2182787.

[16] L. Liu, "The Impact of Mobile Applications on Personalized Learning Paths in Dance Education," International Journal of Interactive Mobile Technologies (iJIM), vol. 19, no. 05, pp. 128–143, Mar. 2025, doi: 10.3991/ijim.v19i05.54525.

[17] S. Mohan, A. Venkatakrishnan, and A. L. Hartzler, "Designing an AI Health Coach and Studying Its Utility in Promoting Regular Aerobic Exercise," ACM Trans Interact Intell Syst, vol. 10, no. 2, Jun. 2020, doi: 10.1145/3366501.

[18] Y. C. Chen, W. W. Liao, M. C. Su, and Y. H. Lin, "Personalized and self-management: Systematic search and evaluation quality factors and user preference of drug reference apps in taiwan," J Pers Med, vol. 11, no. 8, Aug. 2021, doi: 10.3390/jpm11080790.

[19] T. Wang, Y. Xu, X. Zhao, Z. Jiang, and R. Li, "Android malware detection via efficient application programming interface call sequences extraction and machine learning classifiers," IET Software, vol. 17, no. 4, pp. 348–361, Aug. 2023, doi: 10.1049/sfw2.12083.

[20] M. Kaya and Y. Çetin-Kaya, "Seamless computation offloading for mobile applications using an online learning algorithm," Computing, vol. 103, no. 5, pp. 771–799, May 2021, doi: 10.1007/s00607-020-00873-y.

[21] J. Y. Koh, G. W. Peters, I. Nevat, and D. Leong, "Privacy Considerations in Participatory Data Collection via Spatial Stackelberg Incentive Mechanisms," Methodol Comput Appl Probab, vol. 23, no. 3, pp. 1097–1128, Sep. 2021, doi: 10.1007/s11009-020-09798-7.

[22] S. Kumar et al., "Machine learning driven web-based app platform for the discovery of monoamine oxidase B inhibitors," Sci Rep, vol. 14, no. 1, Dec. 2024, doi: 10.1038/s41598-024-55628-y.

[23] D. M. Kotliar and R. Grosglik, "On the Contesting Conceptualisation of the Human Body: Between 'Homo-Microbis' and 'Homo-Algorithmicus,'" Body Soc, vol. 29, no. 3, pp. 81–108, Sep. 2023, doi: 10.1177/1357034X231151855.

[24] V. Feenstra-Kikken et al., "Effectiveness of the HEAR-Aware App for Adults Not Ready for Hearing Aids, but Open to Self-Management Support: Results of a Randomized Controlled Trial," Ear Hear, Nov. 2024, doi: 10.1097/aud.0000000000001533.

[25] Y. Wang et al., "AdaScale: Dynamic Context-aware DNN Scaling via Automated Adaptation Loop on Mobile Devices," Dec. 2024, doi: https://doi.org/10.48550/arXiv.2412.00724.

[26] D. Saisanthiya and P. Supraja, "Heterogeneous Convolutional Neural Networks for Emotion Recognition Combined with Multimodal Factorised Bilinear Pooling and Mobile Application Recommendation," International Journal of Interactive Mobile Technologies (iJIM), vol. 17, no. 16, pp. 129–142, Aug. 2023, doi: 10.3991/ijim.v17i16.42735.

[27] T. E. Sangers et al., "Position statement of the EADV Artificial Intelligence (AI) Task Force on AI-¬assisted smartphone apps and web-¬based services for skin disease," Journal of the European Academy of Dermatology and Venereology, vol. 38, no. 1, pp. 22–30, Jan. 2024, doi: 10.1111/jdv.19521.

[28] A. Chaturvedi et al., "Content Recommendation Systems in Web-Based Mental Health Care: Real-world Application and Formative Evaluation," JMIR Form Res, vol. 7, 2023, doi: 10.2196/38831.

[29] G. Ang and E.-P. Lim, "Learning and Understanding User Interface Semantics from Heterogeneous Networks with Multimodal and Positional Attributes," ACM Trans Interact Intell Syst, vol. 13, no. 3, pp. 1–31, Sep. 2023, doi: 10.1145/3578522.

[30] S. Anwar, M. F. Zolkipli, V. Mezhuyev, and Z. Inayat, "A smart framework for mobile botnet detection using static analysis," KSII Transactions on Internet and Information Systems, vol. 14, no. 6, pp. 2591–2611, Jun. 2020, doi: 10.3837/tiis.2020.06.015.

[31] P. A. Valencia-Londoño, H. Cardona-Rodas, and J. A. Jiménez-Builes, "An Educational Inclusion Model for Adults with Diverse Neuromuscular Conditions through the use of an Artificial Intelligence Algorithm," International Journal of Emerging Technologies in Learning (iJET), vol. 19, no. 05, pp. 35–56, Jun. 2024, doi: 10.3991/ijet.v19i05.46619.

[32] M. Li and L. Niu, "Exercise Recommendation with Preferences and Expectations Based on Ability Computation," Computers, Materials and Continua, vol. 77, no. 1, pp. 263–284, 2023, doi: 10.32604/cmc.2023.041193.

[33] A. N. Sayed, F. Bensaali, Y. Himeur, G. Dimitrakopoulos, and I. Varlamis, "Enhancing building sustainability: A Digital Twin approach to energy efficiency and occupancy monitoring," Energy Build, vol. 328, p. 115151, Feb. 2025, doi: 10.1016/j.enbuild.2024.115151.

[34] A.-J. Wetzel et al., "Ethical, Legal, and Social Implications of Symptom Checker Apps in Primary Health Care (CHECK.APP): Protocol for an Interdisciplinary Mixed Methods Study," JMIR Res Protoc, vol. 11, no. 5, p. e34026, May 2022, doi: 10.2196/34026.

[35] Y. Liu et al., "Towards precision oncology discovery: four less known genes and their unknown interactions as highest-performed biomarkers for colorectal cancer," NPJ Precis Oncol, vol. 8, no. 1, p. 13, Jan. 2024, doi: 10.1038/s41698-024-00512-1.

[36] S. Ha, D. J. Marchetto, S. Dharur, and O. I. Asensio, "Topic classification of electric vehicle consumer experiences with transformer-based deep learning," Patterns, vol. 2, no. 2, p. 100195, Feb. 2021, doi: 10.1016/j.patter.2020.100195.

[37] X. Cheng, N. Li, G. Rysbayeva, Q. Yang, and J. Zhang, "Influence-Aware Successive Point-of-Interest Recommendation," World Wide Web, vol. 26, no. 2, pp. 615–629, Mar. 2023, doi: 10.1007/s11280-022-01055-w.

[38] J. Zhang, Z. Zhao, Z. Cai, Y. Feng, L. Zhu, and Y. Sun, "A Cross Attention Transformer-Mixed Feedback Video Recommendation Algorithm Based on DIEN," Computers, Materials and Continua, vol. 82, no. 1, pp. 977–996, 2025, doi: 10.32604/cmc.2024.058438.

[39] J. H. Yi, W. Kang, S.-E. Kim, D. Park, and J.-H. Hong, "Smart Culture Lens: An Application That Analyzes the Visual Elements of Ceramics," IEEE Access, vol. 9, pp. 42868–42883, 2021, doi: 10.1109/ACCESS.2021.3065407.

[40] L. Yu and Y. Duan, "Responsive and intelligent service recommendation method based on deep learning in cloud service," Front Genet, vol. 13, Nov. 2022, doi: 10.3389/fgene.2022.966483.

[41] A. Mari, A. Mandelli, and R. Algesheimer, "Empathic voice assistants: Enhancing consumer responses in voice commerce," J Bus Res, vol. 175, Mar. 2024, doi: 10.1016/j.jbusres.2024.114566.

[42] M. Haering, M. Bano, D. Zowghi, M. Kearney, and W. Maalej, "Automating the Evaluation of Education Apps With App Store Data," IEEE Transactions on Learning Technologies, vol. 14, no. 1, pp. 16–27, Feb. 2021, doi: 10.1109/TLT.2021.3055121.

[43] I. D. Wahyono et al., "Shared Nearest Neighbour in Text Mining for Classification Material in Online Learning Using Mobile Application," International Journal of Interactive Mobile Technologies, vol. 16, no. 4, pp. 159–168, 2022, doi: 10.3991/ijim.v16i04.28991.

[44] A. Nath, S. Mwesigwa, Y. Dai, X. Jiang, and Z. Zhao, "GENEVIC: GENetic data exploration and visualization via intelli- gent interactive console," Bioinformatics, Oct. 2024, doi: 10.1093/bioinformatics/btae500.

[45] N. Omar, Y. M., and F. A., "Machine Learning Model for Personalizing Online Arabic Journalism," International Journal of Advanced Computer Science and Applications, vol. 11, no. 4, 2020, doi: 10.14569/IJACSA.2020.0110484.

[46] H. Ejaz et al., "Comparison of a Novel Machine Learning–Based Clinical Query Platform With Traditional Guideline Searches for Hospital Emergencies: Prospective Pilot Study of User Experience and Time Efficiency," JMIR Hum Factors, vol. 12, 2025, doi: 10.2196/52358.

[47] J. G. Thanikkal, A. K. Dubey, and M. T. Thomas, "An Efficient Mobile Application for Identification of Immunity Boosting Medicinal Plants using Shape Descriptor Algorithm," Wirel Pers Commun, vol. 131, no. 2, pp. 1189–1205, Jul. 2023, doi: 10.1007/s11277-023-10476-3.

[48] B. CRISTINA, B. CRISTINA, M. BOGDAN, and D. IOANA, "Text Mining in Business. A Study of Romanian Client's Perception with Respect to Using Telecommunication and Energy Apps," Econ Comput Econ Cybern Stud Res, vol. 57, no. 1/2023, pp. 221–234, Mar. 2023, doi: 10.24818/18423264/57.1.23.14.

[49] C. Ning et al., "Development and validation of an explainable machine learning model for mortality prediction among patients with infected pancreatic necrosis," EClinicalMedicine, vol. 80, p. 103074, Feb. 2025, doi: 10.1016/j.eclinm.2025.103074.

[50] J. Xiao et al., "Smart Pharmaceutical Monitoring System With Personalized Medication Schedules and Self-Management Programs for Patients With Diabetes: Development and Evaluation Study," J Med Internet Res, vol. 27, p. e56737, Feb. 2025, doi: 10.2196/56737.

[51] P. Zhou, B. Duan, F. Liu, X. Jing, Y. Lei, and X. Li, "Application of Intelligent Robot in the Intelligent Recommendation of Mobile Application Content," Scalable Computing, vol. 26, no. 2, pp. 708–718, 2025, doi: 10.12694/SCPE.V26I2.3978.

[52] R. Alturki, A. I. Alharbi, S. S. Alqahtani, S. Khan, and V. Truong Hoang, "A Proposed Application for Controlling Overweight and Obesity within the Framework of the Internet of Things," Mobile Information Systems, vol. 2021, 2021, doi: 10.1155/2021/7109099.

[53] E. P. Palacpac et al., "Evaluating the Usability, Perceived Performance, and Perceived Effects of KBGAN iHealth© and KBGAN iFeed© Mobile Apps for Buffalo Management in Selected Municipalities in the Philippines," J Buffalo Sci, vol. 13, pp. 31–45, Feb. 2024, doi: 10.6000/1927-520X.2024.13.04.

[54] S. Teles, C. Paúl, P. Lima, R. Chilro, and A. Ferreira, "User feedback and usability testing of an online training and support program for dementia carers," Internet Interv, vol. 25, p. 100412, Sep. 2021, doi: 10.1016/j.invent.2021.100412.

[55] Q. Hao, K. Zhu, C. Wang, P. Wang, X. Mo, and Z. Liu, "CFDIL: A Context-aware Feature Deep Interaction Learning for App Recommendation," Oct. 27, 2021. doi: 10.21203/rs.3.rs-742592/v1.

[56] N. Rasing, S. Janus, M. Smalbrugge, R. Koopmans, and S. Zuidema, "Usability of an app-based clinical decision support system to monitor psychotropic drug prescribing appropriateness in dementia," Int J Med Inform, vol. 177, p. 105132, Sep. 2023, doi: 10.1016/j.ijmedinf.2023.105132.

[57] X. Zheng et al., "Enhancing Security-Problem-Based Deep Learning in Mobile Edge Computing," ACM Trans Internet Technol, vol. 22, no. 2, May 2022, doi: 10.1145/3458931.

[58] S. Rathee et al., "DILIC: An AI-Based Classifier to Search for Drug-Induced Liver Injury Literature," Front Genet, vol. 13, Jun. 2022, doi: 10.3389/fgene.2022.867946.

[59] M. Xiao, Q. Zhou, L. Lu, X. Tao, W. He, and Y. Zhou, "Applying Deep Learning-Based Personalized Item Recommendation for Mobile Service in Retailor Industry," Mobile Information Systems, vol. 2022, pp. 1–11, Jun. 2022, doi: 10.1155/2022/2364154.

[60] N. Mirbahar, K. Kumar, and A. Ali Laghari, "Enhancing Mobile App Recommendations With Crowdsourced Educational Data Using Machine Learning and Deep Learning," IEEE Access, vol. 13, pp. 38778–38792, 2025, doi: 10.1109/ACCESS.2025.3541116.

[61] C. Zhu et al., "AIM: Automatic Interaction Machine for Click-Through Rate Prediction," IEEE Trans Knowl Data Eng, vol. 35, no. 4, pp. 3389–3403, Apr. 2023, doi: 10.1109/TKDE.2021.3134985.

[62] Y. Ariyanto and T. Widiyaningtyas, "A Systematic Review of Movie Recommender Systems," Journal of Engineering and Technology for Industrial Applications, vol. 10, no. 47, pp. 34–41, Jul. 2024, doi: 10.5935/jetia.v10i47.1052.

[63] M. Kopka, M. L. Schmieding, T. Rieger, E. Roesler, F. Balzer, and M. A. Feufel, "Determinants of Laypersons' Trust in Medical Decision AIDS:

Randomized Controlled Trial," JMIR Hum Factors, vol. 9, no. 2, Apr. 2022, doi: 10.2196/35219.

[64] H. D. Sharma, Y. Misra, SumitKumar, B. M. Rao, and Bhupati, "Expanding an Education-based Collision Detection System Created on Virtual Reality and Augmented Reality," International Journal of Interactive Mobile Technologies, vol. 17, no. 17, pp. 108–120, 2023, doi: 10.3991/ijim.v17i17.42831.

[65] D. Jin, H. Yin, and Y. H. Gu, "Shuffle-PG: Lightweight feature extraction model for retrieving images of plant diseases and pests with deep metric learning," Alexandria Engineering Journal, vol. 113, pp. 138–149, Feb. 2025, doi: 10.1016/j.aej.2024.11.052.

[66] Y. Geng et al., "Identification and validation of platelet-related diagnostic markers and potential drug screening in ischemic stroke by integrating comprehensive bioinformatics analysis and machine learning," Front Immunol, vol. 14, Jan. 2024, doi: 10.3389/fimmu.2023.1320475.

[67] M. R. Rafi, F. Hu, S. Li, A. Song, X. Zhang, and Z. O'Neill, "Deep Weighted Fusion Learning (DWFL)-based multi-sensor fusion model for accurate building occupancy detection," Energy and AI, vol. 17, Sep. 2024, doi: 10.1016/j.egyai.2024.100379.

[68] U. A. Bhatti et al., "Deep Learning-Based Trees Disease Recognition and Classification Using Hyperspectral Data," Computers, Materials & Continua, vol. 77, no. 1, pp. 681–697, 2023, doi: 10.32604/cmc.2023.037958.

[69] S. H. Lee, H. Lee, and J. H. Kim, "Enhancing the Prediction of User Satisfaction with Metaverse Service Through Machine Learning," Computers, Materials and Continua, vol. 72, no. 3, pp. 4983–4997, 2022, doi: 10.32604/cmc.2022.027943.

[70] R. Sinha et al., "Leveraging Genomic Associations in Precision Digital Care for Weight Loss: Cohort Study," J Med Internet Res, vol. 23, no. 5, p. e25401, May 2021, doi: 10.2196/25401.

[71] G. Aly Abdelkader, S. Ngnamsie Njimbouom, T. J. Oh, and J. D. Kim, "ResBiGAAT: Residual Bi-GRU with attention for protein-ligand binding affinity prediction," Comput Biol Chem, vol. 107, Dec. 2023, doi: 10.1016/j.compbiolchem.2023.107969.

[72] Md. S. Uddin, W. Slany, and K. J. Hasan, "On-Device Neural Network for Object Train and Recognition using Mobile," International Journal of Interactive Mobile Technologies (iJIM), vol. 18, no. 12, pp. 99–111, Jun. 2024, doi: 10.3991/ijim.v18i12.47895.

[73] G. Ataguba, M. Alhasani, J. Daniel, E. Ogbuju, and R. Orji, "Exploring Deep Learning–Based Models for Sociocultural African Food Recognition System," Hum Behav Emerg Technol, vol. 2024, no. 1, Jan. 2024, doi: 10.1155/2024/4443316.

[74] G. Shaw Jr et al., "Separating Features From Functionality in Vaccination Apps: Computational Analysis," JMIR Form Res, vol. 6, no. 10, p. e36818, Oct. 2022, doi: 10.2196/36818.

[75] O. Oyebode, J. Fowles, D. Steeves, and R. Orji, "Machine Learning Techniques in Adaptive and Personalized Systems for Health and Wellness," Int J Hum Comput Interact, vol. 39, no. 9, pp. 1938–1962, 2023, doi: 10.1080/10447318.2022.2089085.

[76] T. O. Browne, M. Abedin, and M. J. M. Chowdhury, "A systematic review on research utilising artificial intelligence for open source intelligence (OSINT) applications," Int J Inf Secur, vol. 23, no. 4, pp. 2911–2938, Aug. 2024, doi: 10.1007/s10207-024-00868-2.

[77] J. A. García-Sánchez, L. Chavarría-Báez, and E. F. Ruiz-Ledesma, "Recomendation Systems in Education," in Memorias de la Conferencia Iberoamericana de Complejidad, Informatica y Cibernetica, CICIC, International Institute of Informatics and Systemics, IIIS, Mar. 2025, pp. 90–94. doi: 10.54808/CICIC2025.01.90.

Smart Mobile Apps for Responsible Child Management: A Systematic Literature Review

Daniel Celestino, Harol Medina, Cristian Lara, Nemias Saboya
Facultad de Ingeniería y Arquitectura, Universidad Peruana Unión, Lima, Perú

***Abstract*—Children are increasingly using mobile devices, which raises challenges such as restricting access to inappropriate content, reducing excessive screen exposure, and ensuring safe digital habits. Although various parental control applications exist, most studies focus on isolated aspects such as content filtering or screen time management, with limited integration of artificial intelligence (AI) or consideration of children's cognitive and emotional development. This highlights a research gap that requires a systematic review to consolidate existing evidence and identify best practices. Using the PRISMA methodology, a systematic search was conducted in four databases (Web of Science, Science@Direct, Scopus, and Semantic Scholar). After applying inclusion and exclusion criteria, 29 studies were selected for detailed analysis. Results show that AI-based applications can enhance personalization, improve detection of harmful content, and support parents in establishing healthier digital routines. However, limitations persist, including scarce training datasets, lack of algorithm transparency, and limited assessment of practical effectiveness. This review contributes by mapping current solutions, highlighting strengths and weaknesses, and providing evidence-based insights for researchers, parents, educators, and developers to design safer and more effective child-centered mobile applications.**

***Keywords*—Smart mobile apps; parental control; artificial intelligence; screen time regulation; responsible child management**

I. Introduction

Today, children's use of mobile devices has increased rapidly, mainly due to the digitalization of education, leisure, and social activities. However, this increase puts many parents, educators, and public policymakers on notice, who must ensure that these devices are safe, responsible, and valuable for children's development. According to recent studies, long sessions with screens are associated with problems in the physical and mental health of children, including sleep disorders [1], complex social development [2], or exposure to harmful or risky content [3].

As a result of this panorama, multiple technologies and tools have emerged to try to manage children's use of mobile devices; among these solutions, there are parental control applications, which make it easier for adults to visualize and limit access to unwanted content and control screen time. Some of the applications are beginning to incorporate artificial intelligence (AI) to improve their effectiveness, offering more sophisticated functionalities such as the possibility of personalizing experiences [4] and the proactive identification of digital threats; however, there are still open questions about the effectiveness, accessibility, and security of using this tool.

Nevertheless, a clear gap persists in the existing literature: while prior research has explored isolated aspects such as content filtering, privacy risks, or AI-driven personalization, there is a lack of comprehensive reviews that integrate these dimensions into a holistic framework [3]. This gap limits researchers, educators, and developers from understanding how current technologies collectively address the challenge of responsible child management.

This systematic review aims to assess and analyze the best practices and existing technologies for managing children's access to mobile devices. The general purpose is specified in two specific purposes:

1) Analyze the level of protection offered by applications that prevent children from being exposed to inappropriate content.

2) Evaluate the degree to which artificial intelligence is implemented in applications to manage mobile phone use in children.

This first line of research seeks to understand how current applications address the problem of access to inappropriate content. For example, jobs such as "Hackdroid: Child Safe Browser with Parent Control" and "Angel or Devil? A Privacy Study of Mobile Parental Control Apps" have investigated the content filtering capabilities and blocking methods implemented in these technologies. However, they have also highlighted some of their limitations (e.g., not reaching a high degree of accuracy in detecting dangerous content and the privacy risks that users have [3]. All of this undoubtedly implies improving existing technologies to eradicate this problem and guarantee access to a safe environment for children.

The second specific objective is to implement artificial intelligence in mobile applications to control children's device use. AI has positioned itself as a very effective tool both to personalize user experiences and to optimize parental supervision; to cite an example, the word "AI for Improving Children's Health: A Community Case Study" shows how machine learning algorithms can learn from the needs of children through personalized recommendations and real-time alerts. On the other hand, projects such as "The Sustainable Effect of Artificial Intelligence and Parental Control on Children's Behavior While Using Smart Devices' Apps" demonstrate how AI can help reduce screen time and establish good habits of use of mobile devices. However, the technologies are far from overcoming the problems that Alethea N. Sarac et al. cite as relevant: the lack of data for prediction

training and the need to evaluate transparency and ensure ethics in the use of AI [4].

By explicitly addressing this research gap, the present systematic review contributes to the field by mapping current solutions, highlighting best practices, and identifying key limitations. In doing so, it provides researchers, educators, and developers with evidence-based insights to guide future work and the design of more effective, accessible, and ethically responsible applications for child management.

This systematic review has been designed within this framework to determine best practices, the application of technologies, etc., and provide you with the necessary evidence to carry out future research and developments. By focusing on the analysis of relevant and recent research, we also wanted to contribute to the goal of designing more effective, more accessible, and safer solutions applied by parents, educators, and technology developers. This review aims to meet current needs and anticipate the challenges in an increasingly digitized and interconnected world.

The remainder of this study is organized as follows: Section II states the research questions that guide this systematic review. Section III describes the methodology, including the systematic review process, inclusion and exclusion criteria, and quality assessment. Section IV presents the results obtained from the analysis of the selected studies, directly addressing the research questions. Section V presents the discussion. Finally, Section VI concludes the study by summarizing the main findings, outlining limitations, and suggesting directions for future research.

II. Research Questions

This research focuses on secondary objectives based on the answers to the following research questions:

RQ 1: What impact does the implementation of AI have on parental control applications?

RQ 2: What are the existing technological tools for parental control, and how effective are they in protecting children?

RQ 3: What are the most effective practices for detecting inappropriate content and adjusting mobile device usage rules in children?

RQ 4: What level or techniques of restricting inappropriate content do technological parental control tools offer?

III. Methodology

This study analyzes how children can appropriately and safely use mobile phones and tablets through the evaluation of technological solutions, particularly parental control and AI-based applications. To ensure rigor and transparency, we adopted a systematic review methodology following the PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) guidelines. This approach enables the identification, selection, and synthesis of relevant studies in a replicable manner, while also ensuring that the research gap identified in the Introduction is systematically addressed.

A. Systematic Review Process

The systematic review process was conducted according to PRISMA guidelines. Fig. 1 illustrates the flow diagram of identification, screening, eligibility, and inclusion of studies. A total of 4,985 records were retrieved from four major databases (Web of Science, Science@Direct, Scopus, and Semantic Scholar). After title screening, removal of duplicates, and abstract/full-text evaluation, 29 studies met the eligibility criteria and were included in the final analysis (see Fig. 1).

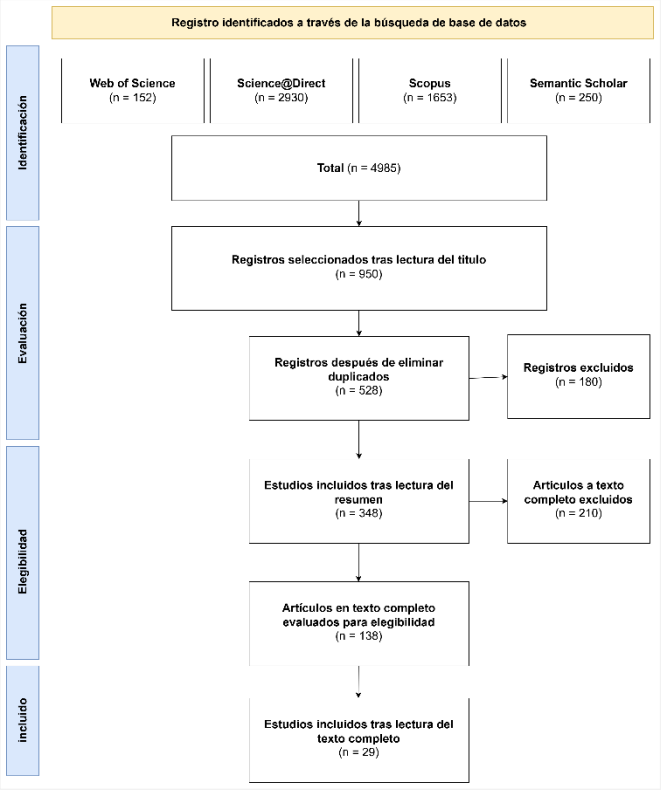


Fig. 1. PRISMA flow diagram showing the process of identifying, selecting, and including studies in the systematic review.

PRISMA Diagram Description:

- Identification: Total number of studies identified from database searches (n = 4985).
- Screening: 950 titles reviewed → 528 records remained after duplicates removed; 180 excluded at abstract level.
- Eligibility: 348 abstracts assessed → 138 full-text articles evaluated with the quality criteria in Table III → 210 excluded.
- Inclusion: The final number of studies included in the review (n = 29).

Although 4,985 records were initially identified, only 29 studies were retained after applying the inclusion and exclusion criteria (see Table I). This reduction reflects the rigorous filtering process designed to ensure that only the most relevant and high-quality studies were analyzed.

B. Inclusion and Exclusion Criteria

The inclusion and exclusion criteria were established to ensure that only studies directly relevant to children's mobile device use, parental control applications, and AI-based solutions were retained. Table I summarizes these criteria.

TABLE I. INCLUSION AND EXCLUSION CRITERIA

Criteria	
Inclusion	***Exclusion***
CI_1 Open access articles.	CE_1 Duplicate items.
CI_2 Articles or studies published in the last 5 years.	CE_2 Items under 5 years
CI_3 Articles addressing mobile device use by young children.	CE_3 Studies that do not address technological solutions or mobile applications.
CI_4 Studies on parental control applications or screen time management systems.	CE_4 Research focused exclusively on adolescents or adults
CI_5 Studies on the effects of mobile device use on children's cognitive and emotional development.	
CI_6 Research that includes AI-based technology or algorithms for the personalization of the use of children's apps.	

Restricting the time frame to the last five years ensured up-to-date coverage of technological advances. Likewise, excluding research on adolescents or adults focused the analysis specifically on children's needs and developmental characteristics.

C. Information Sources

Database

- Web of Science
- Science@Direct
- Scopus
- Semantic Scholar

These sources were selected due to their comprehensive coverage of multidisciplinary research, ensuring both breadth and depth of the literature retrieved.

D. Search Strategy

Search strings were designed to combine three key dimensions: technology terms, child-related terms, and management/parenting terms. The strings were adapted to each database's syntax, as shown below:

WOS ("smart mobile application" OR "mobile app" OR "application" OR "software") AND ("child" OR "youth" OR "kid" OR "minor") AND ("management" OR "monitoring" OR "tracking" OR "supervision") AND ("parenting" OR "caregiver" OR "family" OR "education") AND ("technology" OR "digital" OR "innovation" OR "tools").

Science@Direct ("smart mobile application" OR "mobile app") AND ("child" OR "kid") AND ("monitoring" OR "supervision") AND ("parenting" OR "caregiver").

Scopus ("smart mobile application" OR "mobile app" OR "application" OR "software") AND ("child" OR "youth" OR "kid" OR "minor") AND ("management" OR "monitoring" OR "tracking" OR "supervision") AND ("parenting" OR "caregiver" OR "family" OR "education") AND ("technology" OR "digital" OR "innovation" OR "tools").

Semantic Scholar was used ("smart mobile application" OR "mobile app" OR "application") AND ("child" OR "youth" OR "kid" OR "minor") AND ("management" OR "monitoring" OR "tracking" OR "supervision") AND ("parenting" OR "caregiver" OR "family").

This triangulated strategy maximized sensitivity while maintaining specificity, ensuring a balanced and comprehensive dataset.

E. Study Selection Process

The selection of articles was carried out in two stages:

1) Review of titles and abstracts: Two people reviewed the titles and abstracts to find important articles. When they disagreed, they reached an agreement.

2) Full text review: Items selected in the previous stage were reviewed to determine their final eligibility based on the inclusion criteria.

This two-step procedure ensured transparency and minimized reviewer bias.

F. Data Extraction

A standardized extraction form was used to collect the following information from each study:

- Title, publication year, and source.
- Type of study and target population characteristics.
- Sample size and demographic data.
- Intervention evaluated and type of technology applied (AI, algorithms, parental control features, etc.).
- Methodological approach.
- Main results and effectiveness in reducing risks or exposure.
- Evaluation of AI use, effectiveness of content restriction, and limitations reported.
- Conclusions and recommendations for future research.

G. Evaluating the Quality of the Studies

We reviewed whether the studies were good using a checklist that looked at things like (see Table II):

- Clarity of methodology.
- Description of the target population.
- Effectiveness of parental control tools.
- Evaluation of the implementation of AI or algorithms.

TABLE II. QUALITY ASSESSMENT CHECKLIST

No.	*Question*
QC1	Does the article have a clear and well-defined methodology?
QC2	Does the study clearly describe the target population (parents, children, etc.) and their demographic characteristics?
QC3	Were technology-based parental control tools used effectively in the study?
QC4	Does the study use empirical data, or is it based on reliable qualitative or quantitative evidence?
QC5	Was the implementation of AI or personalization algorithms evaluated in the analyzed applications?
QC6	Does the study include a detailed analysis of the effectiveness of tools for managing screen time in young children?
QC7	Does the article address the impact on risk reduction, such as exposure to inappropriate content or excessive use?
QC8	Does the study provide clear comparisons between different approaches to parental control (manual vs. automated)?
QC9	Does the article adequately describe the limitations of the study?
QC10	Are the results and conclusions well supported by the data presented in the study?

H. Overall Results

At the end of the quality criteria, a total of 29 results were obtained (as shown in Table III), of which those studies analyzed indicated that:

- 85% of the applications reviewed demonstrated significant effectiveness in reducing access to inappropriate content.
- Apps that integrated AI reported a 90% improvement in screen time management compared to manual methods.

TABLE III. QUALITY CRITERIA

Database	***Quality criteria***												
	Articles	QC1	QC2	QC3	QC4	QC5	QC6	QC7	QC8	QC9	QC10	Total (5-10)	Meet
Scopus	Article 1 [5]	1	1	0.5	1	1	1	1	0.5	1	1	9	X
	Article 2 [6]	1	1	0.5	1	0	0.5	0.5	0.5	0.5	0.5	6	X
	Article 3 [7]	0.5	0	1	0.5	1	0.5	1	0.5	0	0.5	5.5	X
	Article 4 [8]	1	1	0	1	0	1	0.5	0	0.5	0.5	5.5	X
	Article 5 [9]	1	1	0.5	1	1	1	1	0.5	0.5	0.5	8	X
	Article 6 [10]	1	1	0.5	0.5	0	1	0.5	1	0.5	1	7	X
	Article 7 [11]	1	1	1	1	1	1	1	0.5	0.5	1	9	X
	Article 8 [12]	1	1	1	0	1	1	1	0.5	1	0	7.5	X
	Article 9 [13]	0.5	0	1	1	0.5	1	1	1	1	1	8	X
Wos	Article 10 [14]	1	1	0.5	1	0	0	0.5	0	1	1	6	X
	Article 11 [15]	1	1	0	1	0	0.5	0	0	1	1	5.5	X
	Article 12 [16]	1	1	0	1	0	0	0.5	0	1	1	5.5	X
	Article 13 [17]	1	1	0	1	0	0	0.5	0	1	1	5.5	X
	Article 14 [18]	1	1	0	1	0	0	0	0	1	1	5	X
Science@Direct	Article 15 [19]	1	1	0	1	1	0.5	0	0	1	1	6.5	X
	Article 16 [20]	1	1	0	1	0	0	1	0	1	1	6	X
	Article 17 [21]	1	0.5	0	1	1	0	0	0	1	1	5.5	X
	Article 18 [22]	1	0.5	0	1	0	0	0	0	1	1	4.5	X
	Article 19 [23]	1	1	0	1	0	0	1	0	1	1	6	X
	Article 20 [24]	1	0.5	0	1	1	0	0	0	1	1	5.5	X
	Article 21 [25]	1	1	0	1	0	0	0	0	1	1	5	X
	Article 22 [26]	1	1	0	1	0	0	0	0	1	1	5	X
	Article 23 [27]	1	1	0	1	0	0	1	0	1	1	6	X
Semantic Scholar	Article 24 [28]	1	1	1	1	0	1	1	1	1	1	9	X
	Article 25 [29]	1	1	0	1	0	0	1	0	1	1	6	X
	Article 26 [30]	1	1	0	1	0	1	1	0	1	1	7	X
	Article 27 [31]	1	1	1	1	0	1	1	1	1	1	9	X
	Article 28 [32]	1	1	0	1	0	1	1	0	1	1	7	X
	Article 29 [33]	1	1	1	1	0	1	1	1	1	1	9	X
TOTAL												190	29

IV. RESULTS

In this part, we will tell you what we discovered after reviewing many studies on apps that teach children to use cell phones and tablets well. We reviewed 29 studies that met the requirements that had been set. Below, I will tell you the most important results.

A. Description of the Included Studies

Ten studies were included in the final review. Most of the studies focused on the following aspects:

- Effectiveness of mobile applications in restricting access to inappropriate content.
- Implementing artificial intelligence (AI) technologies to personalize the user experience.

B. Characteristics of the Study

We included 20 studies that addressed and contributed more to the issue of parental control and screen time management in children. Characteristics of these studies include:

- Target population: Parents or caregivers of children using mobile devices.
- Type of intervention: Mobile parental control applications and screen time management tools.
- Methodology: There are various approaches, including quantitative and qualitative studies.

C. Quality Assessment

After reviewing the PRISMA flow data, rules are used to determine which documents are useful for the investigation. Table III shows the important and reliable studies according to the established criteria. Fig. 2 shows that most articles come from Scopus (36.8%) and Science@Direct (31.6%). The Semantic Scholar and ISI Web of Science are each 15.8% useful. Fig. 3 shows more articles in 2019 and 2021, but less in 2023 and 2024.

D. Results Related to Research Questions

RQ 1: What impact does the implementation of AI have on parental control applications?

The reviewed studies consistently indicate that artificial intelligence enhances parental control applications by providing automated and adaptive features. AI improves the filtering of inappropriate content, recommends age-appropriate material, and alerts parents when risky behaviors are detected [6], [9], [11], [12], [13]. These applications outperform manual parental monitoring by reducing exposure to harmful content and supporting healthier digital habits for children. The main impacts and benefits are summarized in Table IV.

RQ 2: What are the existing technological tools for parental control, and how effective are they in protecting children?

A wide range of technological solutions has been developed to help parents ensure child safety online. These include traditional parental control apps, AI-driven applications, system-embedded functions, and educational tools. While AI tools provide the highest levels of automated protection, educational approaches demonstrate long-term effectiveness by promoting responsible digital autonomy. A detailed comparison of these tools and their effectiveness is presented in Table V.

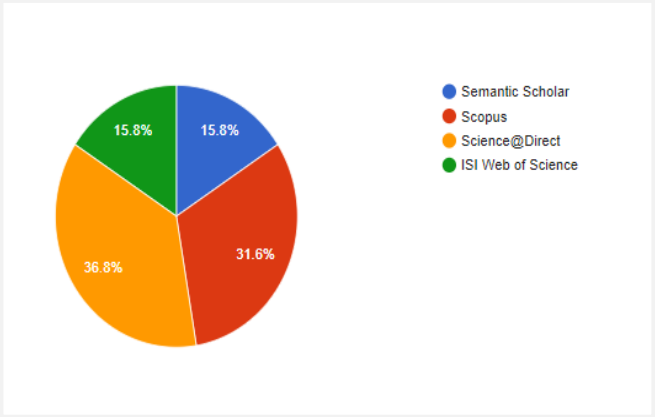


Fig. 2. Distribution percentage of the records identified according to the source of origin.

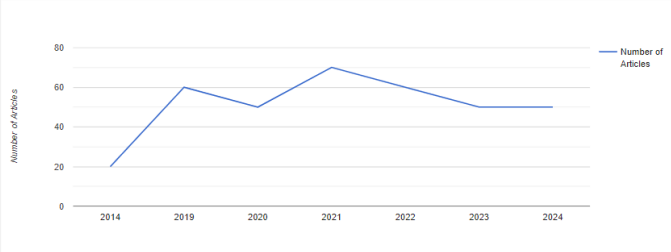


Fig. 3. The trend in the number of articles published between 2014 and 2024.

TABLE IV. IMPACT OF AI ON PARENTAL CONTROL APPLICATIONS

Impact	Benefits	Quote
AI filters content based on the user's age and the sustainability of child behavior.	It prevents children from seeing things they should not.	[11], [13]
AI recommends appropriate educational videos or games for children.	It helps children learn good things and have safe fun.	[9], [11]
AI-powered social media reviews everything before publishing it.	Children can interact online without being exposed to dangerous content.	[11], [13]
The AI detects graphic videos and deletes them or warns the user.	Notify parents or automatically block dangerous content.	[6], [10], [11]
The study says AI is more effective than traditional parental controls.	It is more efficient and reliable than relying on parents alone.	[11], [12]

TABLE V. PARENTAL CONTROL TOOLS

Technological tool	Effectiveness in child safety	Description	Quote
Parental Control Apps	Effective at blocking inappropriate content, but it is dependent on parental involvement.	They allow monitoring and restricting the use of devices.	[6], [10], [12], [17]
Artificial Intelligence (AI)	It is more effective than manual control; it significantly reduces exposure to risky content.	Automatically filters content and suggests age-appropriate material	[11]
Functions embedded in systems	It is useful for beginners but limited in advanced functionalities.	Parental control options are included in operating systems.	[12]
Educational tools	It is very effective in the long term; it promotes digital independence and better habits.	Encourage self-regulation and education about the safe use of technology.	[9], [31]

RQ 3: What are the most effective practices for detecting inappropriate content and adjusting mobile device usage rules in children?

The evidence highlights a combination of technological and behavioral practices. On the technological side, AI-based content filters and behavioral detection algorithms automatically prevent access to harmful material. On the parental side, active supervision, responsible use education, and guided self-management strategies reinforce children's digital resilience [5], [11], [24] .The most effective practices identified are listed in Table VI.

TABLE VI. PRACTICES FOR THE DETECTION OF INAPPROPRIATE CONTENT

Effective Practice	Description	Quote
Using AI to filter content	AI-based tools automatically detect inappropriate content and suggest safe alternatives.	[11], [12]
Behavior detection	Algorithms identify suspicious patterns, such as access to restricted content or interactions with strangers.	[11], [13]
Active parental supervision	Parents review device use and set clear boundaries.	[5], [6]
Responsible use education	Parents teach children to recognize risks and make responsible decisions.	[29], [30], [31]
Open discussions	Parents and children have regular conversations about using technology and the associated risks.	[24]
Guided self-management	Children are allowed to make decisions under initial supervision and constant feedback.	[24]

RQ 4: What level or techniques of restricting inappropriate content do technological parental control tools offer?

The studies analyzed show that parental control tools apply multiple levels of restriction, ranging from automated AI-based filters to user profiles and real-time monitoring. Techniques such as age-based restrictions, app blocking, and activity history reviews are widely used to tailor the digital environment to children's developmental needs [6], [10]. The full set of techniques and their examples are summarized in Table VII.

TABLE VII. TECHNIQUES AND LEVELS OF RESTRICTION

Technique	Description	Example	Quote
Automatic Filtered	Using AI and algorithms to detect and block inappropriate content (words, images, videos)	AI on YouTube Kids filters out content that is not suitable for children.	[5], [6]
Keyword detection	Identification of offensive phrases and replacement with asterisks or warnings.	Big data-based text tools replace offensive phrases with asterisks.	[19], [22]
User profiles	Creation of child profiles to adjust the content according to the child's age.	YouTube Kids and Netflix Kids use AI to show appropriate content for children.	[25], [26]
Age Restrictions	Parents enter the child's age, and the system automatically adjusts the recommendations.	AI controls filter content according to the user's age.	[16]
App Blocking	Blocking inappropriate apps or games for children.	Qustodio and Net Nanny allow specific applications to be blocked.	[14], [27]
Real-time monitoring	Real-time monitoring of children's use of devices.	Screen Time (iOS) allows parents to see which apps their children are using.	[30]
Automatic notifications	Notifications to the parent if the child attempts to access restricted content.	Parental control tools send automatic alerts to parents.	[15], [16], [17]
Activity History	Review of browsing history or use of applications to identify patterns of concern.	Parents review activity history for unusual behaviors.	[5], [6]

E. Bibliometric Analysis

1) Keyword co-occurrence map: The keyword co-occurrence map generated with VOSviewer offers a detailed overview of the thematic relationships in the analyzed scientific literature (see Fig. 4). This map shows 23 elements (keywords), 253 links, and a total connection strength 7194. The visualization uses colors to represent different clusters or thematic groupings, while the size of the nodes reflects the frequency of each keyword's appearance in the articles. The lines connecting the nodes indicate the intensity of the relationship between terms, which allows for identifying significant patterns in the field of study. In addition, the color bar at the bottom of the map indicates the temporal evolution of

the connections between keywords, allowing us to observe how specific topics have gained relevance over time.

One of the emerging core themes is childcare, as evidenced by the prominent presence of the keyword "child". This word is one of the map's most extensive and centralized nodes, suggesting its persistent importance in research. However, when examining the temporary color bar, it can be seen that child-related connections have increased significantly recently, especially from mid-2021 to early 2022. This indicates that research on children has experienced remarkable growth recently due to factors such as the COVID-19 pandemic, the increased use of mobile technologies, and the need to address specific educational and health challenges in this age group.

Another interesting aspect is how the keyword 'child' is strongly interconnected with terms related to mobile technology, such as mobile applications and telemedicine, and educational and healthcare concepts, such as education, healthcare personnel, and self-care. These connections show that current research explores new ways to support children through digital and remote interventions. In addition, terms such as preschool, adolescent, and young adult are closely linked to children, suggesting a broad perspective that addresses different stages of child and youth development.

Finally, the temporal analysis highlights the importance of clinical and methodological studies in child research. Terms such as major clinical study, controlled study, and randomized controlled trial are connected with child, indicating that the scientific community is conducting rigorous and validated research to better understand children's challenges and needs. In addition, terms such as questionnaire and review point to the importance of data collection and the systematic review of previous research, reinforcing the robustness of current findings.

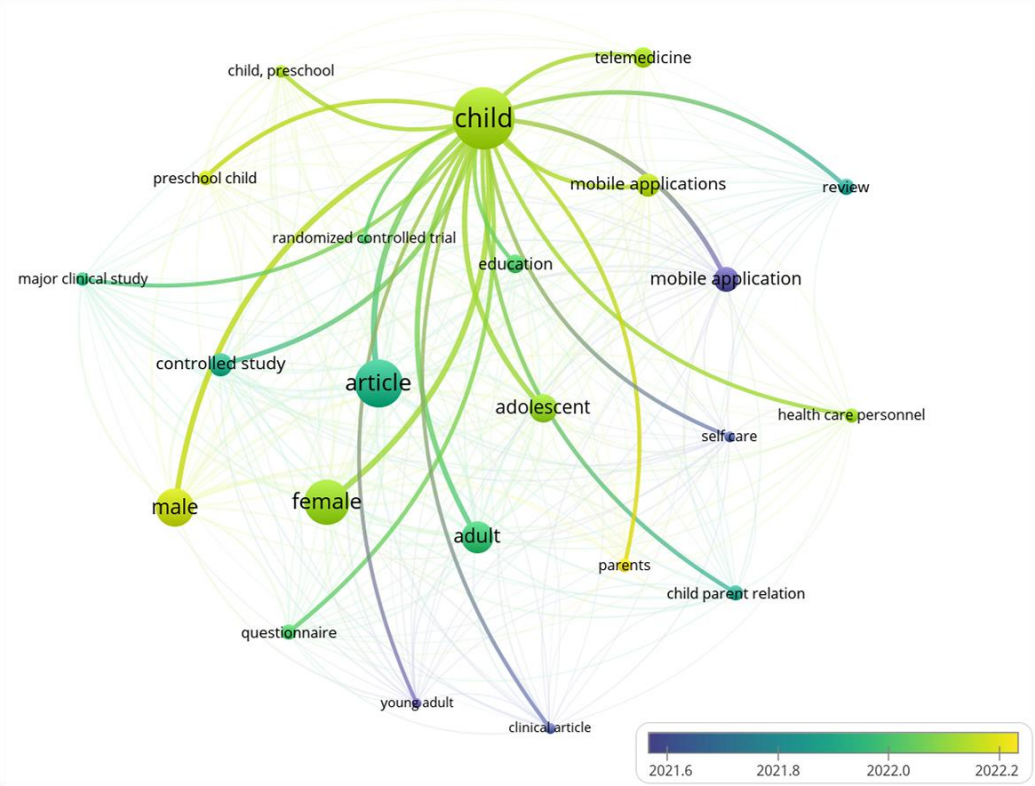


Fig. 4. Keyword co-occurrence map in VOSviewer.

V. Discussion

The results achieved from this systematic review allow us to affirm that the technological tools for parental control are indicated as effective in controlling access to adequate information for children and limiting the time of use of their devices [5]. It is important to note that the final dataset consisted of only 29 studies, despite the identification of nearly 5,000 initial records. This relatively small number results from the strict inclusion and exclusion criteria applied, which removed older publications, studies focusing on adolescents or adults, and those lacking technological or AI-based approaches. While this reduction limited the size of the dataset, it ensured that the review concentrated exclusively on recent, relevant, and methodologically robust evidence. Therefore, the number of included studies should not be seen as a weakness but rather as a reflection of the stringent standards required to maintain rigor and focus in this systematic review. Applications that contain artificial intelligence (AI) are remarkable for providing advanced possibilities for adapting to user limitations and adjusting to how the child uses the application. This adaptation of constraints to use improves the technology experience. It promotes responsible use, vital in a context where children increasingly have access to mobile devices [28]. As summarized in Table V, different parental control tools vary in their effectiveness: while AI-based solutions significantly

reduce exposure to inappropriate content, traditional apps remain dependent on parental involvement, and educational tools show long-term benefits by promoting digital autonomy.

The second aspect of some interest that derives from the studies analyzed is the variety of methodologies used to analyze the validity of these tools. For example, some studies have used surveys of parents to verify that children had less access to inappropriate material; other studies have instead used the analysis of data from the applications themselves to measure the time of use and the manner of children's access to the information. This evident variety of methodologies used explicitly mentions the complexity of the problem and has also shown the need to conceptualize parental control from a set of methodological approaches [29], [30]. As seen in Table II, although most studies describe their methodologies and target populations clearly, fewer provide comprehensive comparisons between manual and automated parental control, showing heterogeneity in methodological rigor.

However, despite the positive results, there are important limitations in the evidence that supports this review. As an example, several of the included studies have a small sample size, which complicates the generalization of the results and their applicability to specific broader contexts [20], [32], [33]. Many studies focused on specific groups of children and their parents, which might not represent the families' experience. The lack of common standards to measure the usefulness of parental control applications also reduced the validity of the reported results, limiting the possibilities of comparing studies and, therefore, being able to draw definitive conclusions. These concerns were addressed from the beginning by applying strict inclusion and exclusion criteria (Table I), which ensured that only recent, open-access, and relevant studies were retained, but still revealed gaps in generalizability.

On the other hand, the review process used has inherent limitations that must be considered. Although rigorous criteria were used to select the articles, relevant works may have been excluded because of the exclusion of specific bases or languages. The selection of the search terms may also have limited the identification of the key research, rejecting articles that could have contributed to this review.

The central question posed by the analysis of mobile applications for the responsible management of access to digital content is the relationship between technology and children's development. Thus, although the applications offer practical solutions to parents, the risks of excessive mobile device use must be explicit. Studies indicate that excessive use of mobile devices can negatively influence children's attention, school performance, and social-emotional development [21]. The techniques identified in this review (Table VII) illustrate how diverse strategies—such as automatic filtering, keyword detection, real-time monitoring, and user profiles—are being implemented to mitigate these risks, although their lack of standardization reduces comparability across studies.

The research results reflect the need not to leave everything available, from management to education, only to parental control applications. Parents must be actively involved in their children's digital education, establishing clear margins and limits and leading a conversation about the responsible use of technology over time. Consequently, the analysis has important conclusions for parents, educators, and app developers. The good news is the social acceptance of the relationship between the use of technology and the management of food access by parents, as well as the integration of technologies based on Artificial Intelligence in managing children's access to digital content through applications. Improvements in these tools could better suit every child and family, most likely translating into greater effectiveness. It is also necessary to focus on those who develop the software to create agile and simple applications used by all families, regardless of the technological capabilities of each parent or caregiver [9], [10].

Future research must be proposed using larger cultural and demographic diversity samples. The richness of this initial research has to be in finding the uses made of these tools in different families and the most interesting characteristics in other contexts; it could also be interesting to check in what way the virtuality of these applications can be effective in the long term, especially about the effects of the tools on the emotional or cognitive development of the child. Investigating how access to digital or non-digital content can change children's behavior and health can yield fascinating information that can be used to refine tools or optimize their use.

VI. Conclusion

Through this work, it has been possible to understand how applications for controversial mobile devices protect minors regarding inappropriate content and device use, thus generating a safer and more responsible digital environment. We found that tools compatible with secure browsers, content filters, or real-time monitoring have made important developments in blocking potentially dangerous or inappropriate content. Existing solutions use innovative techniques, such as geofencing, which allows the entire family to control children's use of their devices or other devices in the family environment. However, we have also found significant issues related to privacy and transparency. Many of these tools handle a large amount of personal data without the explicit consent of users, which raises important ethical and regulatory concerns. This highlights the need to create stricter regulatory frameworks that balance digital protection with respect for minors' rights and ensure their information is not misused.

On the other hand, artificial intelligence has been positioned as an excellent ally in responsibly managing children's mobile device use. AI-based applications can offer advanced functionalities that are complementary to traditional palliative strategies. These include detecting problematic patterns of use, anticipating risks before they have materialized, customizing the rules of use based on the specific behavior of each child, and more. Thus, they allow parents to adopt a more proactive and adaptive role in supervising digital devices. However, these technologies still present important barriers, such as their lack of economic accessibility, the technological infrastructure they require, and the lack of simple tools for families without advanced technical knowledge. In short, although these technologies represent essential advances towards a safer digital environment, there is still work to make them effective, accessible, and ethical so that they can be adopted in the daily

lives of all families without distinction of resources or knowledge.

To validate the findings of this review, the results were contrasted across different methodological approaches, including parent surveys and application usage data analysis, which consistently reported improvements in content filtering and screen time regulation. Furthermore, our synthesis shows convergence between studies using traditional parental control apps and those incorporating AI-based solutions, reinforcing the reliability of the conclusions. Compared with similar reviews in the field, which often focused narrowly on screen time or content filtering, this study provides a more holistic perspective by integrating technological, cognitive, and ethical dimensions. This comparative approach not only validates the robustness of our analysis but also highlights the distinct contribution of this review: the identification of both strengths and persistent gaps in AI-based parental control applications.

REFERENCES

[1] M. Tkaczyk, D. Lacko, S. Elavsky, M. Tancoš, and D. Smahel, "Are smartphones detrimental to adolescent sleep? An electronic diary study of evening smartphone use and sleep," Comput Human Behav, vol. 149, p. 107946, Dec. 2023, doi: 10.1016/J.CHB.2023.107946.

[2] A. Stevic and J. Matthes, "A vicious circle between children's non-communicative smartphone use and loneliness: Parents cannot do much about it," Telematics and Informatics, vol. 64, p. 101677, Nov. 2021, doi: 10.1016/J.TELE.2021.101677.

[3] J. Pimienta et al., "Mobile apps and children's privacy: a traffic analysis of data sharing practices among children's mobile iOS apps," Arch Dis Child, vol. 108, no. 11, pp. 943–945, Nov. 2023, doi: 10.1136/ARCHDISCHILD-2023-325960.

[4] A. Ganju, S. Satyan, V. Tanna, and S. R. Menezes, "AI for Improving Children's Health: A Community Case Study," Front Artif Intell, vol. 3, p. 544972, Jan. 2021, doi: 10.3389/frai.2020.544972.

[5] H. Pratiwi, N. I. Hasanah, S. Purnama, M. Ulfah, and A. Saripudin, "Adaptation to digital parenting in a pandemic: A case study of parents within higher education," South African Journal of Childhood Education, vol. 12, no. 1, 2022, doi: 10.4102/SAJCE.V12I1.1166.

[6] D. Szabó, K. Győri, P. Lajos, and G. Pusztai, "Systematic literature review on parental involvement in digital education," Journal of Media Literacy Education, vol. 16, no. 2, pp. 98–107, 2024, doi: 10.23860/JMLE-2024-16-2-8.

[7] Ş. Türen and P. Bağçeli Kahraman, "The predictive relationships between children's digital game addiction tendencies and mothers' digital parenting awareness and digital literacy levels," Educ Inf Technol (Dordr), Feb. 2024, doi: 10.1007/S10639-024-12932-4.

[8] J. Zhao, B. Duron, and G. Wang, "KOALA Hero: Inform Children of Privacy Risks of Mobile Apps," Proceedings of Interaction Design and Children, IDC 2022, pp. 523–528, Jun. 2022, doi: 10.1145/3501712.3535278.

[9] G. Wang, J. Zhao, M. Van Kleek, and N. Shadbolt, "Protection or Punishment? Relating the Design Space of Parental Control Apps and Perceptions about Them to Support Parenting for Online Safety," Proc ACM Hum Comput Interact, vol. 5, no. CSCW2, Oct. 2021, doi: 10.1145/3476084.

[10] S. Çankaya and H. F. Odabaşi, "Parental controls on children's computer and internet use," Procedia Soc Behav Sci, vol. 1, no. 1, pp. 1105–1109, 2009, doi: 10.1016/J.SBSPRO.2009.01.199.

[11] O. Alrusaini and H. Beyari, "The Sustainable Effect of Artificial Intelligence and Parental Control on Children's Behavior While Using Smart Devices' Apps: The Case of Saudi Arabia," Sustainability (Switzerland), vol. 14, no. 15, Aug. 2022, doi: 10.3390/SU14159388.

[12] V. Gnanasekaran and K. De Moor, "Usability, Security, and Privacy Recommendations for Mobile Parental Control," ACM International Conference Proceeding Series, pp. 138–143, Jun. 2023, doi: 10.1145/3590777.3590800.

[13] J. V. Assis and G. Valença, "Is My Child Safe Online? - On Requirements for Parental Control Tools in Apps used by Children," Journal on Interactive Systems, vol. 15, no. 1, pp. 823–838, Aug. 2024, doi: 10.5753/JIS.2024.4240.

[14] J. M. Muñoz-Rodríguez, P. Torrijos Fincias, S. Serrate González, and A. Murciano Hueso, "Digital environments, connectivity and education: Time perception and management in the construction of young people's digital identity," Revista Espanola de Pedagogia, vol. 78, no. 277, pp. 457–475, 2020, doi: 10.22550/REP78-3-2020-07.

[15] A. M. B. Batista and R. P. Costa, "Qualitative and Visual Methodologies in the Study of Digital Parenting: Methodological challenges," New Trends in Qualitative Research, vol. 9, pp. 286–292, 2021, doi: 10.36367/NTQR.9.2021.286-292.

[16] J. P. Ziker et al., "Parent–Child Adaptive Responses for Digital Resilience," Soc Sci, vol. 14, no. 4, Apr. 2025, doi: 10.3390/SOCSCI14040197.

[17] L. Bertrandias, Y. Bernard, and L. Elgaaied-Gambier, "How Using Parental Control Software Can Enhance Parents' Well-Being: The Role of Product Features on Parental Efficacy and Stress," Journal of Interactive Marketing, vol. 58, no. 2–3, pp. 280–300, May 2023, doi: 10.1177/10949968221144270.

[18] "Problematizing Digital Parenting: Control Practices, Sharenting, and Privacy-All Databases." Accessed: May 05, 2025. [Online]. Available: https://www.webofscience.com/wos/alldb/full-record/WOS:001338879400001

[19] S. Almoallim and C. Sas, "Toward Research-Informed Design Implications for Interventions Limiting Smartphone Use: Functionalities Review of Digital Well-being Apps," JMIR Form Res, vol. 6, no. 4, Apr. 2022, doi: 10.2196/31730.

[20] C. Santos-Silva et al., "Feasibility and Acceptability of Pediatric Smartphone Lung Auscultation by Parents: Cross-Sectional Study," JMIR Pediatr Parent, vol. 7, Jan. 2024, doi: 10.2196/52540.

[21] J. N. P. Nobre et al., "Quality of interactive media use in early childhood and child development: a multicriteria analysis," J Pediatr (Rio J), vol. 96, no. 3, pp. 310–317, May 2020, doi: 10.1016/J.JPED.2018.11.015.

[22] C. N. Pope, E. Sezgin, S. Lin, N. L. Morris, and M. Zhu, "Adolescents' attitudes and intentions to use a smartphone app to promote safe driving," Transp Res Interdiscip Perspect, vol. 4, p. 100090, Mar. 2020, doi: 10.1016/J.TRIP.2020.100090.

[23] Y. Theopilus, A. Al Mahmud, H. Davis, and J. R. Octavia, "Persuasive strategies in digital interventions to combat internet addiction: A systematic review," Int J Med Inform, vol. 195, p. 105725, Mar. 2025, doi: 10.1016/J.IJMEDINF.2024.105725.

[24] V. Gnanasekaran and K. De Moor, "Providing user perspectives to the next generation of parental controls," Int J Child Comput Interact, vol. 44, p. 100735, Jun. 2025, doi: 10.1016/J.IJCCI.2025.100735.

[25] J. van Hoboken and R. Fathaigh, "Smartphone platforms as privacy regulators," Computer Law & Security Review, vol. 41, p. 105557, Jul. 2021, doi: 10.1016/J.CLSR.2021.105557.

[26] K. E. Champion et al., "Parental Information Needs and Intervention Preferences for Preventing Multiple Lifestyle Risk Behaviors Among Adolescents: Cross-sectional Survey Among Parents," JMIR Pediatr Parent, vol. 6, Jan. 2023, doi: 10.2196/42272.

[27] P. A. Busch and S. McCarthy, "Antecedents and consequences of problematic smartphone use: A systematic literature review of an emerging research area," Comput Human Behav, vol. 114, p. 106414, Jan. 2021, doi: 10.1016/J.CHB.2020.106414.

[28] M. Jiménez-Morales, M. Montaña, and P. Medina-Bravo, "Childhood use of mobile devices: Influence of mothers' socio-educational level," Comunicar, vol. 28, no. 64, pp. 19–26, 2020, doi: 10.3916/C64-2020-02.

[29] L. R. Adawiah, Y. Rachmawati, J. Pendidikan, and U. Dini, "Parenting Program to Protect Children's Privacy: The Phenomenon of Sharenting Children on social media," vol. 15, pp. 1693–1602, 2021, doi: 10.21009/JPUD.151.09.

[30] G. Öztürk and F. Şahin Sarıtaş, "Parents' awareness of the conscious use of applications on smart devices: A scale development study," Educ Inf Technol (Dordr), vol. 28, no. 9, pp. 12215–12242, Sep. 2023, doi: 10.1007/S10639-022-11552-0/TABLES/10.

[31] S. Sharma and C. Y. Lee, "Parental mediation and preferences for regulation regarding children's digital media use: role of protection motivation and theory of planned behaviour," Behaviour & Information Technology, vol. 43, no. 8, pp. 1499–1517, Jun. 2024, doi: 10.1080/0144929X.2023.2217275.

[32] R. Onishi, H. Tone, M. Kubota, N. Chino, and F. Maruyama, "Associating Parental Efficacy with the Utility of Smart Devices: A Cross-Sectional Study of Their Role in Alleviating Maternal Parenting Concerns among Infants Aged 6–11 Months," Children, vol. 10, no. 9, Sep. 2023, doi: 10.3390/CHILDREN10091437.

[33] C. C. Chen, W. Y. Lin, M. L. Zou, H. W. Hsu, W. C. Lin, and Y. H. Chen, "The link between smart device use and ADHD—how can reading and physical activity make a difference?," Eur J Public Health, vol. 33, no. Supplement_2, Oct. 2023, doi: 10.1093/EURPUB/CKAD160.837.

DMME-Driven Product Quality Prediction for Semiconductor Manufacturing

Alif Ulfa Afifah, Angga Prastiyan, Fahmi Arif, Fadillah Ramadhan
Industrial Engineering, Institut Teknologi Nasional, Bandung, Indonesia

***Abstract*—Defective products in manufacturing can be reduced by accurately predicting quality outcomes based on process parameters. This study proposes a quality prediction framework for semiconductor manufacturing using the Data Mining Methodology for Engineering Applications (DMME). This study extends DMME with domain-specific preprocessing and demonstrates its superiority on the SECOM dataset compared to other classifiers. Experimental results show that the Random Forest algorithm achieved the highest performance, with 92.99% accuracy and an F-measure of 0.9637, confirming the effectiveness of the proposed approach. These findings highlight the potential of structured, engineering-oriented data mining to improve product quality and support informed decision-making in complex manufacturing environments.**

***Keywords*—Data mining; quality prediction; DMME; semiconductor manufacturing; random forest**

I. INTRODUCTION

Manufacturing refers to the industrial process of converting raw materials into products with higher economic value, whether in the form of finished goods or intermediate components [1]. Semiconductor production is a highly complex stage that involves multiple subprocesses and generates large volumes of process data, which must be effectively controlled and analyzed to improve product quality [2]. Even with continuous advancements in production technology and management practices, product defects still occur in manufacturing processes [3]. Breakdowns in production machinery can further disrupt manufacturing flows and result in substantial losses, emphasizing the importance of preventive maintenance strategies to ensure production stability and consistency in product quality [4]. A practical way to reduce these defects is to analyze production data and pinpoint the process parameters that have the most significant impact on product quality. Understanding these factors enables manufacturers to develop predictive models that help anticipate product outcomes and support informed decision-making for future production runs [5]. When properly built, these models can be more accurate and efficient than traditional inspection-based approaches [6].

Several studies have examined the prediction of quality in various manufacturing contexts. In wafer dicing, for example, models have been shown to predict failures with up to 75% accuracy [7]. In the process of coke production, regression techniques have been employed to map coal properties into performance indicators, such as the coke reactivity index (CRI) and the coke strength after reaction (CSR) [8]. On the other hand, battery production has adopted various methods to model product quality across complex and highly variable production lines [9].

In semiconductor manufacturing, cascade quality prediction techniques that integrate principal component analysis (PCA) with decision tree algorithms, such as ID3, have shown strong results, achieving prediction accuracies of up to 90.02% [10]. While this highlights the promise of predictive modeling in the manufacturing sector, the effectiveness of such models can vary significantly depending on the application area and the characteristics of the data being used.

Reliable quality prediction depends on more than just selecting a good algorithm. The way the problem is defined and the data are prepared play a significant role in the outcome. Quality management, at its core, is about improving process performance and minimizing variation [11], [12]. Predictive models built on process data support this goal by providing a structured, data-driven means of identifying patterns that affect product outcomes [13]. Data mining plays a key part in this process. It combines statistical and machine learning techniques to uncover useful patterns and insights from complex data sources [14], [15], [16]. One common framework for this is the Cross-Industry Standard Process for Data Mining (CRISP-DM), which organizes data mining into six defined phases, from understanding the business context all the way to implementation. Still, CRISP-DM is a general-purpose framework, and it doesn't fully address the needs of engineering-specific problems. To bridge that gap, the Data Mining Methodology for Engineering (DMME) was developed to extend CRISP-DM by incorporating additional steps more suited to technical domains [17].

Despite the availability of various quality prediction methods, many are developed and tested under specific conditions or on a limited set of data. Because of differences in data types, distributions, and feature counts, a model that works well in one context might not deliver the same performance in another. This limitation is especially relevant in semiconductor manufacturing, where the data is often highly complex and challenging to work with. The benchmark that is often used in this space is called the SECOM dataset. It contains many numerical attributes with a nominal label to indicate product quality. While this dataset offers considerable potential for analysis, limited research has explored which machine learning techniques are best suited for handling its specific characteristics, such as high dimensionality and class imbalance. Moreover, there has been little examination of how individual process variables impact prediction outcomes across different modeling approaches.

To fill this research gap, the study focuses on developing a quality prediction model specifically suited to datasets like SECOM. A model that consists of numerical features and categorical labels. The goal is to identify a modeling approach that offers high accuracy while factoring in the distinct difficulties and characteristics of the semiconductor manufacturing process.

The remainder of this study is organized as follows: Section II reviews related works on data-driven quality prediction and outlines the research gap addressed in this study. Section III details the proposed methodology based on DMME. Section IV presents the dataset used in this research, data preparation techniques, describes the modelling approach and selected algorithms. Section V reports the experimental results and Section VI discusses their implications. Finally, Section VII concludes the study with key findings and suggestions for future research.

II. Related Work

Recent advancements have demonstrated the increasing role of machine learning (ML) and data-driven approaches in enhancing semiconductor manufacturing processes. A comprehensive review of ML applications for semiconductor process optimization was presented by [18], in which predictive models, virtual metrology, and advanced process control were highlighted as methods that have improved yield and quality across various stages of production. While these approaches have demonstrated significant improvements, most focus on specific techniques or stages rather than employing a holistic methodology to systematically address data challenges in manufacturing. This study addresses this gap by leveraging the Data Mining Methodology for Engineering Applications (DMME) to develop a comprehensive quality prediction framework for semiconductor processes. In [19], the authors systematically reviewed automated defect inspection using convolutional neural networks (CNNs) on scanning electron microscope (SEM) images, emphasizing the potential of ML to replace traditional, labor-intensive inspection methods. In specific process applications, [20] developed a machine-learning-based prediction model to optimize plasma etching parameters, demonstrating that domain-specific ML techniques can enhance process stability and output quality. Beyond improving accuracy, [21] integrated explainability methods into manufacturing quality prediction models, enabling better transparency and trust in model-driven decisions. Furthermore, [22] addressed a critical challenge of ML adoption in manufacturing by analyzing the effects of data quality and class imbalance on predictive performance, recommending robust preprocessing pipelines for reliable outcomes.

In addition to algorithmic improvements, recent works have explored leveraging modern computing infrastructures to enhance quality management systems. In [23], the authors demonstrated the feasibility of integrating data mining techniques such as Decision Tree, k-Nearest Neighbor, Naïve Bayes, and Random Tree into a Software-as-a-Service (SaaS) platform, enabling low-cost and accessible quality prediction for manufacturing environments. Similarly, [24] proposed a Cloud-based Quality Analyzer (CQA) that incorporates real-time analysis, automated feedback loops, and reduced reliance on human quality engineers, aligning with Industry 4.0 principles of cyber-physical integration and data-driven decision-making. These studies highlight the growing trend toward combining advanced predictive models with cloud-based and intelligent platforms to achieve scalable, efficient, and adaptive quality management solutions.

Despite these contributions, many existing approaches either focus on specific manufacturing tasks or rely heavily on singular algorithmic solutions without a structured methodology to handle data quality issues systematically. To bridge this gap, the present study employs the Data Mining Methodology for Engineering Applications (DMME), which integrates domain-specific preprocessing, feature selection, and algorithm comparison to develop a robust product quality prediction framework tailored for high-dimensional, imbalanced semiconductor manufacturing data.

III. Methodology

This research uses the Data Mining Methodology for Engineering Applications (DMME), which builds on the widely recognized CRISP-DM framework and adapts it specifically for engineering challenges [25]. DMME introduces important technical extensions to the process, including feature selection, dimensionality reduction, and techniques to address class imbalance, which are critical steps due to the high-dimensional and unbalanced nature of the SECOM dataset analyzed here. For example, principal component analysis (PCA) was considered to simplify the dataset. On the other hand, methods like SMOTE were explored to address the uneven distribution between defective and acceptable products. For the classification task, the study focused on five algorithms, i.e., Decision Tree, Random Forest, Support Vector Machine, k-Nearest Neighbor, and Naive Bayes. These were selected because they offer a good mix of transparency, speed, and effectiveness for structured data typically found in manufacturing. More complex options, such as XG Boost or neural networks, were set aside for this work in favor of models that are easier to interpret and commonly used in process engineering. All analyses were conducted using Rapid Miner Studio and Microsoft Excel, ensuring a systematic evaluation of how various process parameters affect product quality in semiconductor manufacturing.

IV. Data Understanding and Methodology

This stage describes the process of collecting and selecting data to be processed for research. The process of predicting product quality requires historical data owned by the company to be used as a reference in the prediction process. The data used must have elements that directly affect the quality of the product or can be referred to as attributes. The data studied in this study were defect data in semiconductor products [25]. The semiconductor dataset can be seen in Table I.

The data above displays several columns of identity information, including the time of manufacture of the product, the product number in the second column, the third column, which contains parameters with both numeric and real characteristics, and the last column, which shows the quality results of the product. There are 590 types of parameters, each with a unique value in the column. The last column in the table indicates the quality of the product, with an accept/reject

classification status characterized by binomial characteristics. The dataset used has multivariate characteristics, with the value of each attribute having numerical characteristics. The dataset is in the manufacturing area. The dataset used contains attributes with numeric characteristics, comprising 591 columns and 1567 rows of production data. This semiconductor dataset accounts for missing values and bias resulting from disturbances in the data collection process, with varying intensities depending on the features of each product. This null value must be considered when preprocessing data or when the algorithm is applied. The data is represented in a raw text file, with each line corresponding to an individual and a feature in the process.

TABLE I SEMICONDUCTOR DATASET

Time	Prod	Attribute				Accept/Reject
		0	*1*	*…*	*589*	
19/07/2008 11:55	1	3030,9	2564,0	…		Accept
19/07/2008 12:32	2	3095,8	2465,1	…	208,2	Accept
19/07/2008 13:17	3	2932,6	2559,9	…	82,9	Reject
19/07/2008 14:43	4	2988,7	2479,9	…	73,8	Accept
19/07/2008 15:22	5	3032,2	2502,9	…	73,8	Accept
…	…	…	…	…	…	…
17/10/2008 6:07	1567	2944,9	2450,8	…	137,8	Accept

A. Data Preparation

The data preparation process involves adjusting the type of data to be used in conjunction with the character data from the dataset. The data prepared for the data type adjustment process is part of the attribute column, where each parameter value has special characteristics that can be classified as a particular data type. An explanation of the data preparation stage is provided below.

1) Data cleansing: The raw dataset, before the cleansing process is carried out to eliminate missing values, can be seen in Table II.

TABLE II SEMICONDUCTOR RAW DATASETS

Time	Prod	Attribute				Accept/Reject
		0	*1*	*…*	*589*	
19/07/2008 11:55	1	3030,9	2564,0	…		Accept
19/07/2008 12:32	2	3095,8	2465,1	…	208,2	Accept
19/07/2008 13:17	3	2932,6	2559,9	…	82,9	Reject
19/07/2008 14:43	4	2988,7	2479,9	…	73,8	Accept
19/07/2008 15:22	5	3032,2	2502,9	…	73,8	Accept
…	…	…	…	…	…	…
17/10/2008 6:07	1567	2944,9	2450,8	…	137,8	Accept

The data cleansing process, which involved eliminating missing values in this study, was performed using Microsoft Excel on the rows and columns of the dataset. After the process of removing missing values is completed in Microsoft Excel, the data can be viewed in Table III.

TABLE III SEMICONDUCTOR DATASETS AFTER CLEANSING DATA FOR MISSING VALUE

Attribute				Accept/Reject
0	*1*	*…*	*589*	
2932,6	2559,9	…	82,9	Reject
2988,7	2479,9	…	73,8	Accept
2946,3	2432,8	…	44,0	Accept
3030,3	2430,1	…	44,0	Accept
3058,9	2690,2	…	95,0	Accept
…	…	…	…	…
3246,3	2499,8	…	23,6	Accept

Then, the data cleansing process is carried out to handle outlier values in the semiconductor dataset using the auto-cleansing feature in RapidMiner Studio software. An overview of auto cleansing can be seen in Fig. 1.

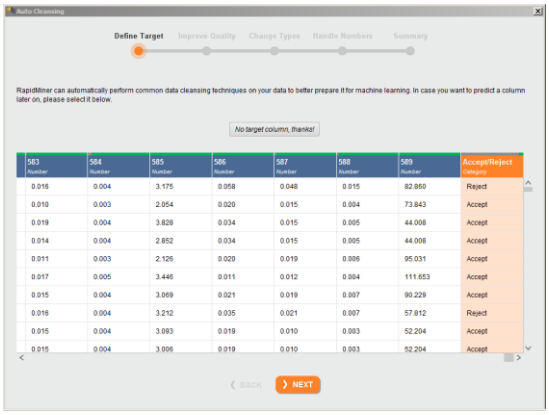


Fig. 1. Cleansing data for outlier handling.

After the auto cleansing process is done, the data after the process can be seen in Table IV.

TABLE IV SEMICONDUCTOR DATASETS AFTER CLEANSING DATA FOR OUTLIERS

Attribute				Accept/Reject
0	*1*	*…*	*589*	
2932,6	2559,9	…	82,9	Reject
2988,7	2479,9	…	73,8	Accept
2946,3	2432,8	…	44,0	Accept
3030,3	2430,1	…	44,0	Accept
3058,9	2690,2	…	95,0	Accept
…	…	…	…	…
3246,3	2499,8	…	23,6	Accept

2) Data transformation: Before the data transformation process is carried out, the dataset to be studied is prepared in advance. Data transformation is performed using a read operator in Rapid Miner, which can be used for various data extensions, including CSV, Excel, URL, SAS files, and several other file extensions that resemble databases. Researchers in this study used an Excel file type (xlsx), so that the operator used was Read Excel. Four steps must be taken when importing data to this operator. First, select files; second, determine the format to be used; third, determine the rows and columns; and finally, select labels to predict. The determination of the label is based on the classification results expected in the study. The prediction process carried out refers to the classification results to be achieved. The processed elements consisted of 392 attribute variables and the outcome of the decision variable, which was labeled. The type of data used depends on the value in each attribute column. In this research dataset, all values in the attribute columns are numeric data types, and the label variable, resulting from the accept/reject decision, is nominal.

B. Modelling

The modeling process is carried out by extracting the data that has been prepared in advance. The extraction process is carried out with the selected algorithms, namely, random forest, decision tree, k-NN, support vector machine, and naïve Bayes. The extraction processes up to the selection of Rules can be seen below.

1) Data extraction: The data extraction stage on the prepared dataset utilizes several operator processes to facilitate the data extraction process. Extraction begins by reading the data from the dataset using the Read Excel operator and then proceeds through the preprocessing process (data preparation) until it reaches the cross-validation process.

2) Cross-validation: Cross-validation is a statistical method for evaluating and comparing algorithms by dividing the dataset into two classifications, namely the training set and the testing set. Validation in this study employs a 10-fold cross-validation approach, performing iterations with 10 data waves that alternate as test data. Cross-validation is used because the validation process emphasizes the use of test data interchangeably, ensuring that the data used to create a model is similar to the original characteristics. The training set is part of the dataset that is used to train the model, while the testing set is a subset of the dataset that is used to evaluate its performance. The principle of the cross-validation process is to conduct repeated experiments (iterations) on the test and train sets. The number of these iterations can be determined according to the intended results. The extraction stage involves several supporting operators that optimize the data to be extracted, ensuring that at the time of cross-validation, the data is relevant for validation.

V. Results

A. Selected Algorithm

The accuracy value of each algorithm used can be seen in Table V.

TABLE V ALGORITHM ACCURACY

Algorithm	Precision	Recall	Accuracy	F *Measure*	G-*Mean*
Random Forest	92.99%	100.00%	92.99%	0.9637	0.4649
k-NN	99.72%	93.21%	92.99%	0.9636	0.4648
SVM	99.91%	92.98%	92.90%	0.9632	0.4645
Decision Tree	98.70%	93.64%	92.55%	0.9610	0.4621
Naive Bayes	93.63%	21.88%	25.97%	0.3547	0.1024

The Rapid Miner results recapitulation for algorithms that have been used in graphical form can be seen in Fig. 2.

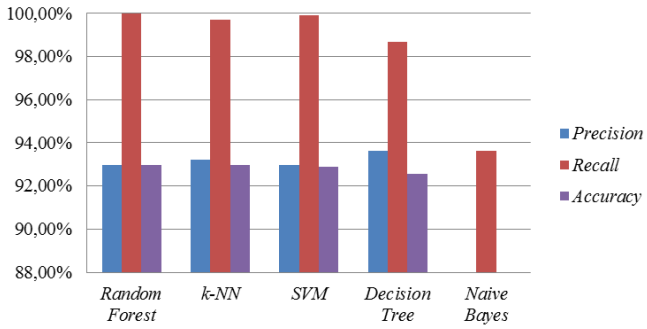


Fig. 2. Comparison of algorithms.

Table V presents the performance metrics for each algorithm. Random Forest achieved the highest accuracy (92.99%) and F-measure (0.9637), outperforming all other classifiers. While Random Forest provided the best performance, its computational cost was higher than single-tree models, which may impact real-time applications.

The five models used have their characteristics based on their work specifications. Based on this research, the random forest model has the highest accuracy and F-measure value, at 92.99% for accuracy and 0.9637 for the F-measure. Therefore, it was selected for use on the dataset. However, this model has weaknesses in terms of computation time. The random forest algorithm uses random attributes and classifies the trees that are formed. The tree building is done recursively based on the data associated with the same label. Tree splitting is performed to divide the data based on the type of attribute used. The Naïve Bayes algorithm classifies labels using probability and statistical methods. The SVM algorithm pursues the optimal hyperplane by maximizing the distance between classes. A hyperplane serves as a separator between classes. Another classification algorithm is the k-Nearest Neighbor algorithm, which classifies new nodes by assigning a value based on the category with the highest number of nearest k neighbors. Finally, the decision tree algorithm serves as the basis for the random forest algorithm, which converts data into decision rules to form a tree, enabling

decision makers to better interpret solutions to the problems that arise.

There is a visualization of the random forest algorithm, which consists of several decision tree diagrams. An illustration of one of the first decision tree visualizations from the Random Forest algorithm, is demonstrated in Fig. 3, following the knowledge mining process already conducted.

The decision tree model in the random forest algorithm can be interpreted as a set of rules, making it easier to understand. Rules are a concise summary of descriptive algorithms that enable the deduction of solutions to a problem, making them easy to understand. One of the rules for the decision tree shown in Fig. 4 is evident.

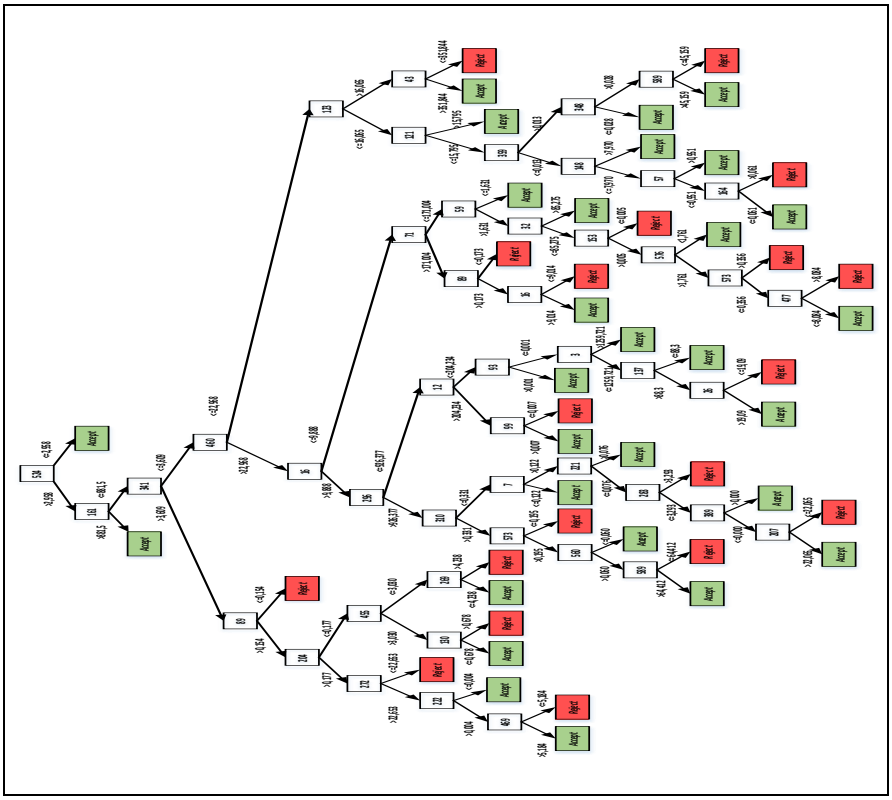


Fig. 3. One of the random forest model diagram visualization.

524 > 2.958
| 161 > 8811.500: Accept {Reject=0, Accept=86}
| 161 ≤ 8811.500
| | 341 > 3.609
| | | 89 > 0.154
| | | | 204 > 0.177
| | | | | 272 > 22.653
| | | | | | 222 > 0.004
| | | | | | | 469 > 5.184: Accept {Reject=0, Accept=10}
| | | | | | | 469 ≤ 5.184: Reject {Reject=2, Accept=0}
| | | | | | 222 ≤ 0.004: Accept {Reject=0, Accept=57}
| | | | | 272 ≤ 22.653: Reject {Reject=2, Accept=0}
| | | | 204 ≤ 0.177
| | | | | 455 > 3.030
| | | | | | 130 > 0.678: Reject {Reject=8, Accept=0}
| | | | | | 130 ≤ 0.678: Accept {Reject=0, Accept=4}
| | | | | 455 ≤ 3.030
| | | | | | 269 > 4.238: Reject {Reject=2, Accept=0}
| | | | | | 269 ≤ 4.238: Accept {Reject=0, Accept=10}
| | | 89 ≤ 0.154: Reject {Reject=5, Accept=0}
| | 341 ≤ 3.609
| | | 460 > 22.968
| | | | 16 > 9.888
| | | | | 296 > 926.377
| | | | | | 310 > 0.331
| | | | | | | 573 > 0.195
| | | | | | | 93 ≤ -0.001
| | | | | | | | 3 > 1259.721: Accept {Reject=0, Accept=22}
| | | | | | | | 3 ≤ 1259.721
| | | | | | | | | 137 > 88.300
| | | | | | | | | | 26 > 1.909: Reject {Reject=5, Accept=1}
| | | | | | | | | | 26 ≤ 1.909: Accept {Reject=0, Accept=3}
| | | | | | | | | 137 ≤ 88.300: Accept {Reject=0, Accept=8}
| | | | 16 ≤ 9.888
| | | | | 71 > 171.004
| | | | | | 89 > 0.173
| | | | | | | 16 > 9.014: Accept {Reject=0, Accept=12}
| | | | | | | 16 ≤ 9.014: Reject {Reject=2, Accept=0}
| | | | | | 89 ≤ 0.173: Reject {Reject=4, Accept=0}
| | | | | 71 ≤ 171.004
| | | | | | 59 > 1.631
| | | | | | | 32 > 85.275: Accept {Reject=0, Accept=52}
| | | | | | | 32 ≤ 85.275
| | | | | | | | 153 > 0.005
| | | | | | | | | 576 > 1.761
| | | | | | | | | | 573 > 0.356: Reject {Reject=5, Accept=1}
| | | | | | | | | | 573 ≤ 0.356
| | | | | | | | | | | 477 > 8.084: Reject {Reject=1, Accept=1}

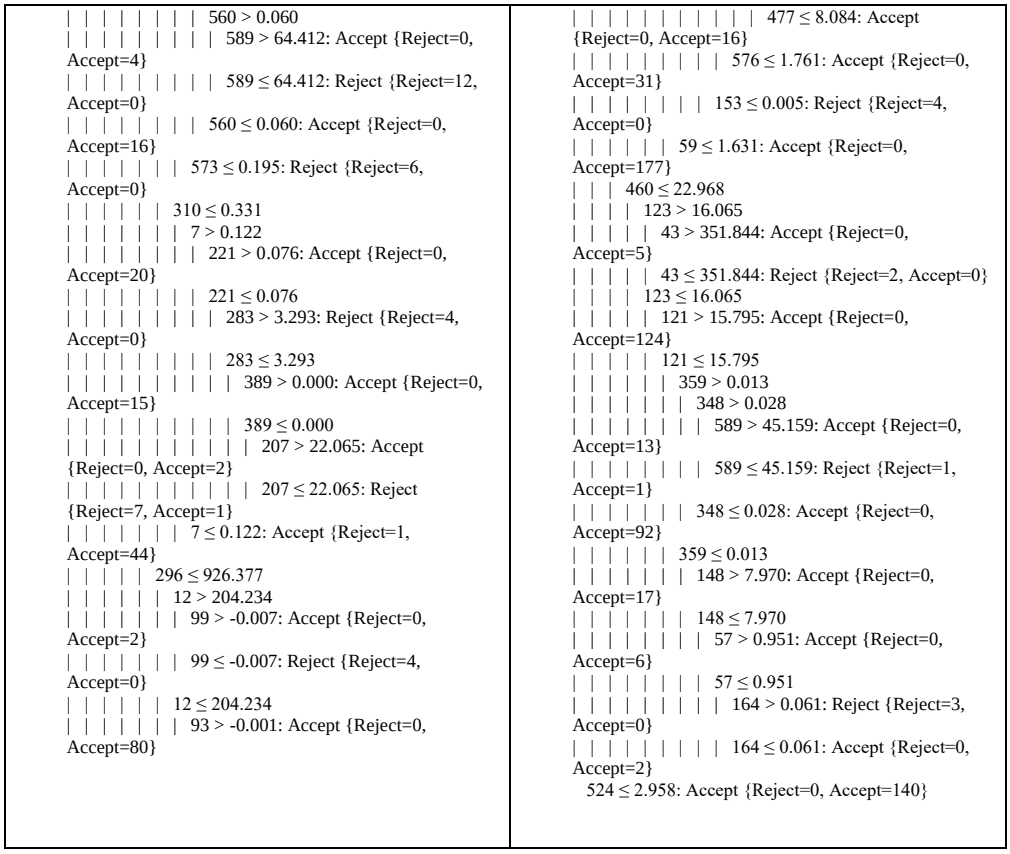


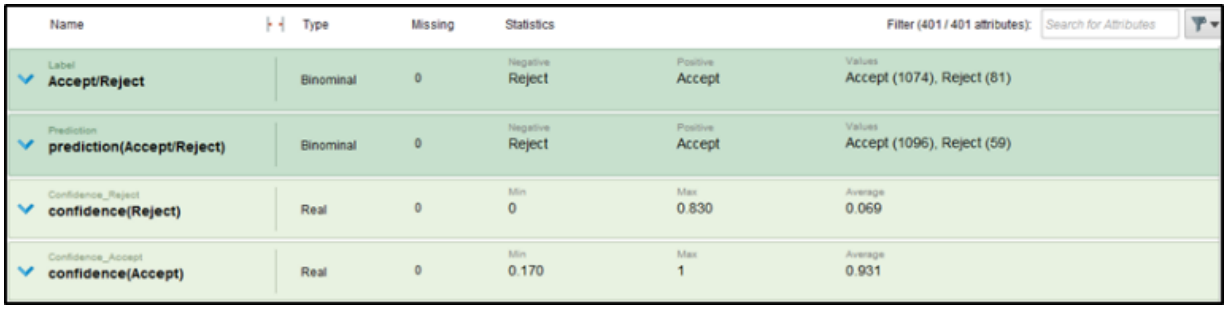
Fig. 4. Rules in the random forest model.

The formulation of the quality prediction model, performed on semiconductor data sets, produces rules with the highest accuracy level, specifically the random forest algorithm, which achieves an accuracy value of 92.99% and an F-measure of 96.37%. This is supported by some knowledge from the results of data extraction regarding the relationship between process parameters and the final quality of the product, as represented by red and green, as shown in Fig. 5.

The results of the quality prediction model are close to the accepted results from the previous production, with a total of 1096 units of products and a failure rate of 59 units. The predicted results are shown in Fig. 6.

Accept/Reject	prediction(Accept/Reject)	confidence(Reject)	confidence(Accept)	0	1	2	3	4	5	6
Reject	Reject	0.600	0.400	2932.610	2559.940	2186.411	1698.017	1.510	100	95.488
Accept	Accept	0.067	0.933	2988.720	2479.900	2199.033	909.793	1.320	100	104.237
Accept	Accept	0.008	0.992	2946.250	2432.840	2233.367	1326.520	1.533	100	100.397
Accept	Accept	0.002	0.998	3030.270	2430.120	2230.422	1463.661	0.829	100	102.343
Accept	Accept	0	1	3058.880	2690.150	2248.900	1004.469	0.788	100	106.240
Accept	Accept	0.100	0.900	2967.680	2600.470	2248.900	1004.469	0.788	100	106.240
Accept	Accept	0.150	0.850	3016.110	2428.370	2248.900	1004.469	0.788	100	106.240
Reject	Reject	0.614	0.386	2994.050	2548.210	2195.122	1046.147	1.320	100	103.340
Accept	Accept	0.007	0.993	2920.070	2507.400	2195.122	1046.147	1.320	100	103.340
Accept	Accept	0.101	0.899	3051.440	2529.270	2184.433	877.627	1.467	100	107.871
Reject	Reject	0.612	0.388	2963.970	2629.480	2224.622	947.774	1.292	100	104.849
Accept	Accept	0.083	0.917	2988.310	2546.260	2224.622	947.774	1.292	100	104.849
Accept	Accept	0.083	0.917	3028.020	2560.870	2270.256	1258.456	1.395	100	104.808
Accept	Accept	0	1	3032.730	2517.790	2270.256	1258.456	1.395	100	104.808
Accept	Accept	0	1	3040.340	2501.160	2207.389	962.532	1.204	100	104.031

Fig. 5. Important factor.



Name	Type	Missing	Statistics		
Label: Accept/Reject	Binominal	0	Negative: Reject	Positive: Accept	Values: Accept (1074), Reject (81)
Prediction: prediction(Accept/Reject)	Binominal	0	Negative: Reject	Positive: Accept	Values: Accept (1096), Reject (59)
Confidence_Reject: confidence(Reject)	Real	0	Min: 0	Max: 0.830	Average: 0.069
Confidence_Accept: confidence(Accept)	Real	0	Min: 0.170	Max: 1	Average: 0.931

Fig. 6. Results of the quality prediction model.

VI. Discussion

The Random Forest algorithm achieved the highest performance among all evaluated classifiers, with an accuracy of 92.99% and an F-measure of 0.9637 on the SECOM dataset. This superior performance can be attributed to Random Forest's ensemble learning approach, which mitigates overfitting and effectively handles high-dimensional data with complex inter-feature dependencies.

The variation in performance across datasets indicates that the proposed Random Forest-based model is particularly effective for high-dimensional, structured process data like SECOM. For datasets with fewer features or lower variance, simpler models such as SVM or k-NN may suffice. This suggests that the algorithm's suitability is strongly linked to data characteristics, reinforcing the importance of feature engineering in DMME. These insights confirm that structured methodologies like DMME can deliver practical, high-performance solutions in real-world manufacturing contexts.

These findings highlight the value of adopting a structured methodology that integrates domain-specific preprocessing, feature selection, and model comparison, as facilitated by DMME. By systematically addressing data quality issues, the framework ensures that the most suitable algorithm is selected for the given manufacturing context. Future research could extend this work by incorporating cost-sensitive learning, real-time adaptation, and integration with cloud-based quality management systems to further improve predictive performance in dynamic production environments.

VII. Conclusion

This study applied the Data Mining Methodology for Engineering Applications (DMME) to predict product quality in semiconductor manufacturing. By integrating advanced preprocessing techniques and comparing multiple classification algorithms, the framework effectively addressed the challenges of high dimensionality and data imbalance in the SECOM dataset. Experimental results identified Random Forest as the most suitable classifier, achieving 92.99% accuracy and an F-measure of 0.9637.

The findings confirm that enhancing traditional CRISP-DM with domain-specific preprocessing and algorithm evaluation improves predictive performance in complex manufacturing environments. This work contributes to the development of structured, data-driven methodologies for quality prediction and provides a practical approach for industries seeking reliable, interpretable models. Future research will explore broader manufacturing applications, incorporate cost-sensitive learning, and integrate this framework into real-time, multi-stage production systems to further strengthen defect prevention and decision-making.

References

[1] Hajo Wiemer, Lucas Drowatzky, Steffen Ihlenfeldt, 'Data Mining Methodology for Engineering Applications (DMME) – A Holistic Extension to the CRISP-DM Model, Chair of Machine Tools Development and Adaptive Controls, Institute of Mechatronic Engineering, Technische Universität Dresden, 01069 Dresden, Germany. 2019.

[2] Pedro Espadinha-Cruz, Radu Godina, Eduardo M. G. Rodrigues, 'A Review of Data Mining Applications in Semiconductor Manufacturing', A Review of Data Mining Applications in Semiconductor Manuacturing', Processes, 9(2), 305, 2021.

[3] Xinmin Zhang, Manabu Kano, Masahiro Tani, Junichi Mori, Junji Ise, Kohhei Harada, "Prediction and causal analysis of defects in steel products: Handling nonnegative and highly overdispersed count data", Control Engineering Practice, Volume 95, 2020.

[4] Alif Ulfa Afifah, Fifi Herni Mustofa, Ainun Zahra Mustika, 'Determining Preventive Maintenance Schedule on Press Machine Components using Age Replacement Methods', E3S Web of Conferences 484, 01024, 2023.

[5] R. W. House and T. Rado, "An approach to artificial intelligence," in *IEEE Transactions on Communication and Electronics*, vol. 83, no. 70, pp. 111-116, Jan. 1964.

[6] E. Demirbilek and J. -C. Grégoire, "Machine learning based reduced reference bitstream audiovisual quality prediction models for realtime communications," *2017 IEEE International Conference on Multimedia and Expo (ICME)*, Hong Kong, China, pp. 571-576, 2017.

[7] Te-Jen Su, Yi-Feng Chen, Jui-Chuan Cheng, Chien-Liang Chiu, "An artificial neural network approach for wafer dicing saw quality prediction", Microelectronics Reliability, Volume 91, Part 2, 2018.

[8] Lauren North, Karen Blackmore, Keith Nesbitt, Merrick R. Mahoney, Models of coke quality prediction and the relationships to input variables: A review, Fuel, Volume 219, Pages 446-466, 2018.

[9] Sebastian Thiede, Artem Turetskyy, Arno Kwade, Sami Kara, Christoph Herrmann, Data mining in battery production chains towards multi-criterial quality prediction, CIRP Annals, Volume 68, Issue 1, Pages 463-466, 2019.

[10] Fahmi Arif, Nanna Suryana, Burairah Hussin, Cascade Quality Prediction Method Using Multiple PCA+ID3 for Multi-Stage Manufacturing System, IERI Procedia, Volume 4, Pages 201-207, 2013.

[11] Joseph m. Juran, Joseph A. De Feo, Juran's Quality Handbook: The Complete Guide to Performance Excellence, McGraw Hill, 2010.

[12] Peter D. Mauch, Quality Management Theory and Application, CRC Press, Boca Raton, 2009.

[13] Roger Sauter, Introduction to Engineering Statistics and Six Sigma, Technometrics, Vol 49, 2007.

[14] Jiawei Han, Micheline Kamber, Jian Pei, Data Mining: Concepts and Techniques. 2012.

[15] Efraim Turban, Jay E. Aronson, Ting Peng Liang, Decision Support Systems and Intelligent Systems, Prentice Hall. 2003.

[16] M. Kantardzic, Data Mining: Concepts, Models, Methods & Algorithms. 2011

[17] Steffen Huber, Hajo Wiemer, Dorothea Schneider, Steffen Ihlenfeldt, DMME: Data mining methodology for engineering applications – a holistic extension to the CRISP-DM model, Procedia CIRP, Volume 79, Pages 403-408, 2019.

[18] Ying-Lin Chen, Sara Sacchi, Bappaditya Dey, et al. Exploring Machine Learning for Semiconductor Process Optimization: A Systematic Review. TechRxiv. July 16, 2024.

[19] Thibault Lechien, Enrique Dehaerne, Bappaditya Dey, Victor Blanco, Sandip Halder, Stefan De Gendt, Wannes Meert, 'Automated Semiconductor Defect Inspection in Scanning Electron Microscope Images: a Systematic Review', arxiv, Cornell University, 2023.

[20] Changmin Kima, Seunghwan Leeb, Muyoung Kima, Min Sup Choic, Taesung Kimb, and Hyeong-U Kima , 'Machine Learning-Based Prediction of Atomic Layer Control for MoS2 via Reactive Ion Etcher', Applied Science and Convergence Technology 2023; 32(5): 106-109, 2023.

[21] Dennis Gross, Helge Spieker, Arnaud Gotlieb, Ricardo Knoblauch, 'Enhancing Manufacturing Quality Prediction Models through the Integration of Explainability Methods', arxiv. Cornell University, 2024.

[22] Jiarui Xie, Lijun Sun, Yaoyao Fiona Zhao, 'On the Data Quality and Imbalance in Machine Learning-based Design and Manufacturing—A Systematic Review', Engineering, 2024.

[23] Fahmi Arif, Fadillah Ramadhan, Wildan Sayf, 'Data Mining for Quality Prediction in Software-as-A-Service Concept: A Case Study in Offset Printing Company', Quality Innovation Prosperity/Kvalita Inovacia Prosperita 27/3, 2023.

[24] Fahmi Arif and Isa Setiasyah Toha, 'Enhancing manufacturing quality system using cloud-based quality analyzer', AIP Conference Proceedings 2772, 080012, 2023.

[25] McCann, M. and Johnston, A, UCI Machine Learning Repository [http://archive.ics.uci.edu/ml/datasets/SECOM]. Irvine, CA: University of California, School of Information and Computer Science, 2018.

AIoT-Based Waste Classification for Solid Waste Management to Accomplish the SDGs

T. M. Shien[1], M. Batumalay[2], Balasubramaniam Muniandy[3], Pavan Kumar Pagadala[4], Vinoth Kumar. P[5]
Faculty of Data Science & Information Technology (FDSIT), INTI International University, Nilai, Malaysia[1, 2, 3]
Centre for Data Science and Sustainable Technologies, INTI International University, Nilai, Malaysia[2]
Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Hyderabad-500075, Telangana, India[4]
Sri Krishna College of Engineering and Technology, Coimbatore, India[5]

***Abstract*—The Fourth Industrial Revolution, known as IR4.0 technologies, has enhanced global economic capabilities and productivity, but there are negative environmental and health impacts due to industrialisation and urbanisation, such as greenhouse gas emissions and global warming. One effective method that reduces environmental impact is to conduct a waste classification program that incorporates the principles of 3R. The proposed work includes educating individuals and businesses on the importance of waste reduction, promoting reusable products and packaging, and implementing effective recycling systems. Additionally, Governments could also incentivise sustainable practices through tax breaks or invest in renewable energy sources to reduce greenhouse gas emissions associated with industrial processes. The proposed study aims to develop automated waste classification technology that can help reach SDGs 11, 12, and 13 by making waste management more efficient, increasing recycling and resource recovery rates, and cutting down on greenhouse gas emissions. The proposed system is developed using a deep learning algorithm with a microprocessor and microcontroller managing sensors and actuators to perform waste sorting based on the classification result. This distinguishes the proposed system from existing manual and RFID-based approaches by integrating AIoT with a user incentive mechanism, improving both accuracy and public adoption. This technology enhances overall sustainability and promotes a more circular economy by enabling the reuse and recycling of materials for their own well-being through process innovation.**

***Keywords*—Solid waste management; waste classification; artificial intelligence of things (AIoT); well-being; process innovation; sustainable development goals (SDG); recycling; circular economy**

I. Introduction

One of the most pressing environmental pollution issues that most countries face in the twenty-first century is the damage that solid waste pollution causes to the environment [1]. Waste emissions have substantially increased at a visible rate, making waste management more challenging [2]. Over 64% of the population in developing countries and 84% of the population in developed countries will be expected to be living in urban areas by the year 2050. This urbanisation trend is expected to result in a global waste generation of approximately 3.4 billion metric tonnes, which is a significant rise compared to the current 2.01 billion metric tonnes [3–4].

Solid waste pollution not only leads to the spread of biological diseases and localised odours but also exerts a significant impact on the soil, water, and air [5]. However, the nature of solid waste can also cause environmental pollution that has an impact on health issues. For instance, the common infectious diseases are, namely, typhoid and cholera, and non-communicable diseases such as cancer, asthma, or birth defects in babies [6]. According to estimates, waste pollution is the main cause of death in nations with environmental pollution issues.

Malaysia relies significantly on landfills, which offer a more cost-effective and straightforward waste disposal solution compared to other methods [7]. Landfilling and incineration have traditionally been considered cost-effective approaches for solid waste disposal, but both methods inflict severe damage on the environment [8]. When significant amounts of solid waste are either disposed of or incinerated in landfills, the decomposition process releases potent greenhouse gases, such as carbon dioxide and methane, which significantly contribute to global warming and climate change [9]. Despite the availability of more advanced technologies for incineration and other waste disposal methods, landfilling remains the predominant approach for waste disposal in all countries. Based on current trends and future projections, it is anticipated that waste generation will continue to increase and that unsustainable waste disposal techniques like landfilling and incineration will remain in use. As a result, the harm that solid waste causes to the ecological balance will only get worse [10]. This rapid growth in solid waste production exceeds the capacity for timely disposal, resulting in a significant amount of waste remaining unmanaged and contributing to the proliferation of illegal landfills or open dumps [11].

Unfortunately, the unprecedented emergence of the coronavirus disease 2019 (COVID-19) has further intensified the adverse environmental consequences of solid waste, as the global population has resorted to increasing utilisation of single-use plastic cutlery and personal protective equipment (PPE) to curb the spread of the virus [12]. Consequently, the management of municipal solid waste (MSW) faces significant challenges due to the generation of additional substantial quantities of clinical waste, including masks, PPE, batteries, and empty oxygen cylinders [12]. Since the outbreak began, an estimated 1.6 million metric tonnes of plastic waste are produced globally every day, while 3.4 billion single-use masks or face shields are thrown away every day [13]. Furthermore, the transmission of

the COVID-19 virus occurs through sneezing, coughing, and touching surfaces contaminated with COVID-19, which results in the need for additional precautions in the disposal of municipal solid waste to avoid unnecessary virus transmission or environmental contamination [14–15]. Nevertheless, current waste management systems remain largely manual, labour-intensive, and expose workers to health risks, especially during pandemics such as COVID-19. This research addresses these gaps by proposing an automated AIoT-based system, combining deep learning image recognition with IoT-enabled hardware and a user incentive mechanism.

The recycling of waste to mitigate greenhouse gas emissions has always been the primary objective in managing municipal solid waste (MSW) in Malaysia [16]. For instance, the government implemented various measures aimed at improving waste management and encouraging recycling in the 12th Malaysia Plan (2021 to 2025). These measures included banning the use of single-use products, such as plastics and packaging materials, in restaurants and public events. Additionally, the construction of comprehensive scheduled waste treatment and disposal facilities was prioritised to combat illegal dumping. These initiatives aim to achieve a waste recycling rate of 35% by the end of 2025. Furthermore, Malaysia has made a commitment to decrease its greenhouse gas emissions intensity by 45% from 2005 levels by 2030 upon signing the Paris Agreement [17]. Although the government has implemented many frameworks for sustainable solid waste management, the main problem lies in the lack of alternatives to waste management and the poor implementation of waste policies [18]. Moreover, Malaysia stands among the countries significantly contributing to environmental pollution and health issues resulting from waste accumulation due to the irresponsible and ignorant attitudes of consumers towards waste disposal [19]. According to the research results, many residents are aware of the negative effects that waste has on the environment, but they nonetheless frequently engage in behaviours such as dumping waste in rivers or accumulating it with other waste [20].

II. Literature Review

One of the main goals of the SDGs, according to the United Nations' 2030 Agenda for Sustainable Development, is to address the significant ecological and economic challenges that the entire world faces [21]. Furthermore, the management of municipal solid waste is encompassed within the framework of the Sustainable Development Goals (SDGs) and the Paris Agreement on Climate Change [22]. For instance, SDG 11 (Sustainable Cities and Communities) promotes the establishment of efficient and sustainable waste management systems in urban areas, leading to reduced pollution and cleaner cities. Additionally, SDG 12 (Responsible Consumption and Production) aims to limit pollution resulting from excessive solid waste decomposition in the environment through the principles of 3R (reduction, recycling, and reuse). Lastly, SDG 13 (Climate Action) focuses on decreasing greenhouse gas emissions associated with waste production and disposal to mitigate environmental pollution caused by existing inefficient waste management systems.

As shown in Fig. 1, the proposed system flow integrates sensors, actuators, and cloud services for automated waste classification, using waste classification techniques employing deep learning and IoT technology. The suggested solution tackles the issue and meets the 3R (reduce, reuse, recycle) principles and the Sustainable Development Goals (SDGs). This makes waste collection and categorization possible. In addition, depending on the kind and quantity of waste collected, the system would update the users' membership points or e-wallet points in a cloud database. Residents are encouraged to actively engage in waste recycling activities by means of this incentive.

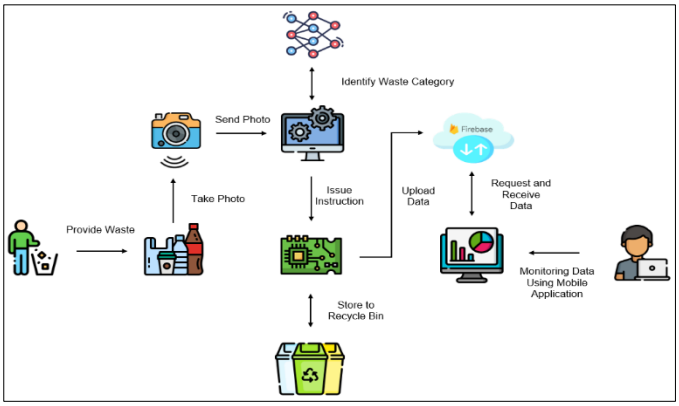


Fig. 1. Project system flow.

A trash classification system that is automated and increases accuracy and efficiency in waste management can be easily created with the help of current AIoT technologies. Through the utilisation of artificial intelligence and the Internet of Things, the system can efficiently recognise and classify various waste materials, thereby lessening the need for manual effort and increasing the overall rate of recycling. The analysis of the data gathered from this system to find patterns and trends also enables better decision-making regarding waste management techniques, making 3R recycling rates easier.

In order to develop an automated waste classification system and raise 3R recycling rates, the primary goal of the project is to research current AIoT technologies. Subsequently, create a system for classifying waste and create mobile applications to encourage recycling and raise user knowledge of recycling. Lastly, assess the accuracy and dependability of the suggested system.

III. Methodology

A well-defined procedure is used in the software development life cycle (SDLC) to generate and produce a high-quality product [22]. The waterfall model is regarded as one of the simplest approaches, among others. For software or system development, the waterfall model comprises six separate phases: requirements, design, development, testing, deployment, and maintenance. However, only the requirement analysis, design, development, and testing phases of the waterfall model have been carried out. The research approach was applied in three steps. Data collection, system development, and testing are the first three phases. Phase 1: Data Collection thus covers requirements, analysis and design tasks, Phase 2: System Development covers development tasks, and Phase 3: Testing

covers testing tasks. The waterfall strategy is consistent with this study methodology [23].

A. Phase 1: Data Collection

First, information was gathered through an interview and a questionnaire. A method for gathering quantitative data was using questionnaire, and a means for gathering qualitative data was an interview. In-depth perspectives and comments on the proposed system, as well as the current waste classification system, were obtained from Malaysians using both methods. While the interviews allowed open-ended responses that offered deeper insights and viewpoints, the questionnaire included closed-ended questions to collect quantifiable data on preferences and opinions. Combining the two approaches could result in a thorough grasp of the current system as well as possible upgrades. To make sure that the planned system would satisfy the requirements and preferences of its intended users, this data gathering phase was essential in providing guidance for the project's design and development phases.

Through the questionnaire, 80 respondents were Malaysian, including 50 males and 30 women. Based on Fig. 2, the most commonly used and recycled solid waste types among the respondents were aluminium and plastic cans. Fig. 3 shows that respondents were in complete agreement about the contribution of trash sorting to environmental harm. Furthermore, the bulk of respondents mentioned how difficult it is to physically separate waste. The data from the questionnaire will be crucial in informing strategies for waste management and environmental conservation in Malaysia. It is evident that there is a need for more efficient waste sorting systems to address the challenges highlighted by the respondents.

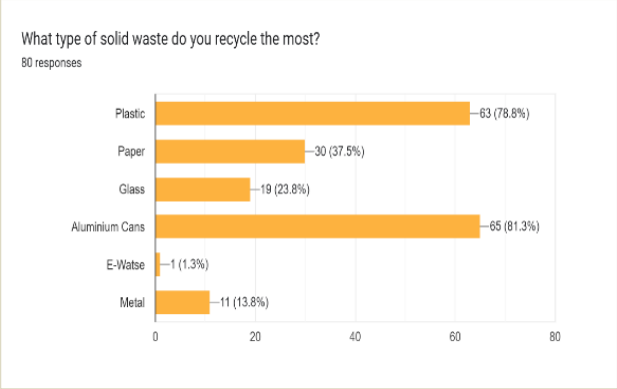


Fig. 2. Type of solid waste recycled the most.

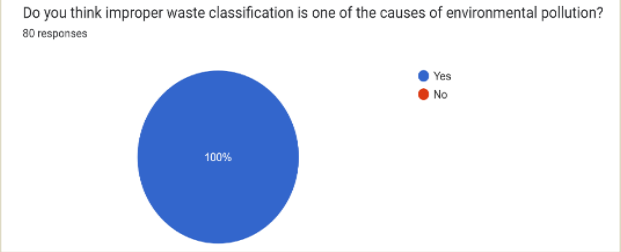


Fig. 3. Perception of waste misclassification causing environmental pollution.

When it comes to recycling, Fig. 4 illustrates whether consumers have segregated solid waste by kind. A mere 21.3% of respondents sorted their rubbish prior to recycling, whereas only 78.8% of respondents did not separate their waste, according to the statistics. The survey investigated the reasons behind not sorting waste for those participants who replied "no" to the preceding question. Fig. 5 demonstrates that while 20.6% of respondents said they knew nothing about trash separation, 22.2% mentioned being extremely busy, and 56.6% thought the separation process was difficult. Only a small percentage said it took a lot of time.

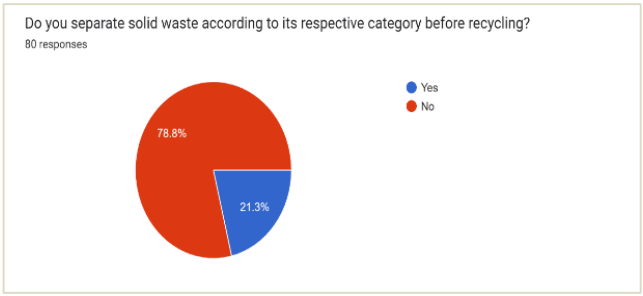


Fig. 4. The response to solid separation.

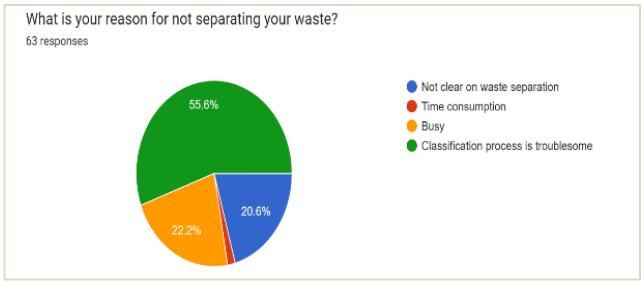


Fig. 5. The reason for not separating the trash.

According to the respondents, the suggested system will classify garbage more accurately than current approaches, as seen in Fig. 6. Based on the pie chart, 92.5% of respondents think that the intelligent garbage classification system is superior to the manual sorting system currently in use, while 7.5% disagree. In addition, respondents think the system can stop viruses from spreading, particularly COVID-19, and only 19 out of 80 respondents think it can increase recycling rates.

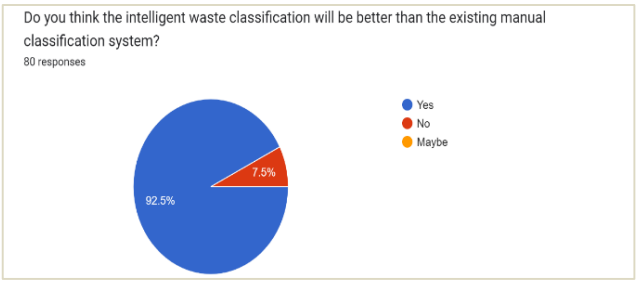


Fig. 6. Feedback on the intelligent waste classification.

As seen in Fig. 7, respondents' readiness to pay for the proposed system reveals that 71.3% found fees between RM1 and RM1,000 acceptable, while 26.3% would not pay any fees at all. Lastly, the purpose of this query is to determine which system components respondents value the most. According to

Fig. 8, the user interface (38.7%), the hardware design (10%), and 51.2% of respondents, the functionality of the suggested system is the most important feature. The responses to the questionnaire gave important information on the status of waste classification as it stands today as well as the respondents' reluctance to move forward with it. Regarding the respondents' opinions on the proposed system, it is likewise evident. The questionnaire indicates that opinions of the proposed waste classification system are largely positive.

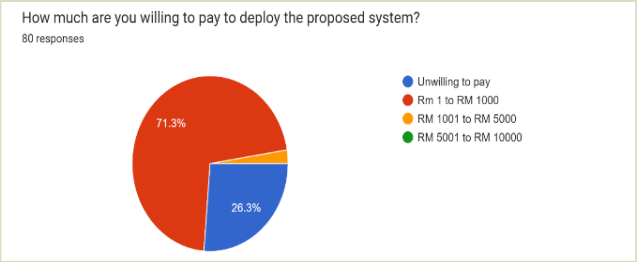


Fig. 7. The willingness of respondents to pay for the proposed system.

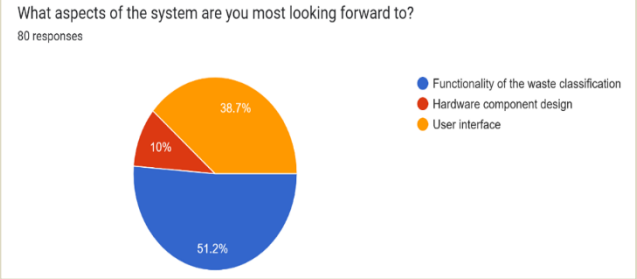


Fig. 8. The expectation on the system.

To ascertain whether the public appendices and papers in the report were acceptable to them, five individuals were interviewed. Table I enumerates the interview questions along with their intended use. It was intended to ascertain the respondents' awareness of existing waste management procedures and assess their opinions regarding waste generation in Malaysia. In addition, the interview aims to shed light on respondents' perceptions of the relationship between incorrect trash recycling and contamination of the environment, as well as suggest possible solutions. to find out how the respondents felt about environmental pollution and what steps they thought could be taken to stop waste pollution. Furthermore, information was gathered regarding respondents' perceptions of environmental pollution as well as suggested or plausible countermeasures for waste pollution. The main ideas of the suggested system and the respondents' assessment of it were clear to them.

The interview summaries indicate that the participants conveyed concerns about Malaysia's high rates of waste generation and the shortcomings of current environmental protection strategies. Inappropriate waste classification has been demonstrated to be one of the primary sources of environmental pollution. It was found that raising public awareness and educating people about recycling trash was crucial for finding a solution to this issue. The proposed garbage sorting system received positive feedback, with time and effort savings predicted. The system has to be more precise and easier to use, according to respondents, in order to increase user numbers, increase recycling rates, and raise awareness of environmental issues.

According to the interview summaries, participants expressed apprehensions regarding Malaysia's elevated waste generation rates and the insufficiency of existing methods for safeguarding the environment. One of the main causes of pollution in the environment has been shown to be improper waste classification. It was determined that increasing public awareness and education about waste recycling was essential to solving this problem. Positive input was given to the suggested garbage sorting system, with time and effort savings anticipated. Respondents did, however, highlight the need for the system to be more accurate and user-friendly to draw in more users, boost recycling rates, and promote environmental consciousness.

Next, one of the Unified Modelling Language (UML) diagrams that illustrates the functionality and extent of the system is the use case diagram, which illustrates the interaction between the actors and various use cases [24]. The case diagram provided below illustrates the relationships between actors and use cases in the proposed system.

Fig. 9 depicts the use case for the Solid Waste Classification System, which illustrates the interactions among six main actors.

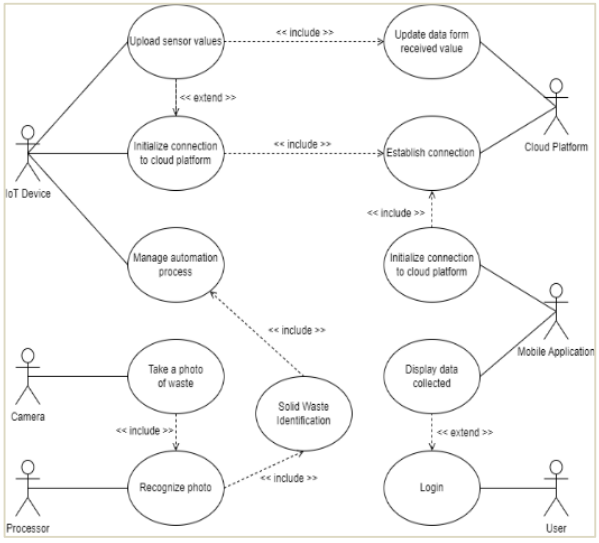


Fig. 9. Use case diagram for the solid waste classification system.

The Internet of Things (IoT) device is one of these characters: it oversees managing the actual automated garbage sorting procedure and uploading sensor readings to the cloud; Camera: takes and uploads trash photos to the processor. Processor: using a neural network-trained deep learning model to analyse the acquired photos and issue commands based on the classification results; Cloud Platform: a location for managing and storing all collected data; Mobile Application: a smartphone app that tracks data by connecting to a cloud database; User: to confirm their identification, access their account. The waste's weight is converted into points by the system after it has identified and categorised the waste category. These points are then updated in the user's account.

B. Phase 2: System Development

The system was created using Android Studio, Firebase, Arduino IDE, Thorny on Raspberry Pi, and Google Collaborative. Thorny and the Arduino IDE employ Python and C++ to manage the hardware components of the Raspberry Pi and Arduino, while Google Colab is used for free to train and run the deep learning model. The necessary database is managed by Firebase, and Android Studio is used to create the Android application.

The deep learning model was trained using a set of TensorFlow Keras libraries using the MobileNet architecture, a rapid and lightweight deep learning architecture designed for embedded devices with limited computational resources, such as the Raspberry Pi. The MobileNet architecture is optimised for image classification tasks. The MobileNet v1 neural network model was trained using RGB pictures with a pixel resolution of 224x224. It's a total of 28 layers, including batch normalisation, depth-wise convolutional, ReLU activation, and convolutional layers. With the addition of GlobalAveragePooling2D and two thick custom layers, the total number of layers rises to 31. With 97.23% accuracy on the training set and 99.24% accuracy on the validation set, the model demonstrated exceptional accuracy during training. The model's ability to accurately classify photographs was proven by its impressive 98.98% accuracy with a low loss of 0.0499 on the test set. The model has previously undergone pre-training, and the parameters are established.

The primary functions of the Raspberry Pi 4B are image capture and processing, while the Arduino Uno is used to communicate with actuators and sensors to obtain sensor readings and trigger tangible actions. As shown in Fig. 10, the hardware connection schema integrates Raspberry Pi, Arduino Uno, Pi Cameras, sensors, and actuators to support automated waste classification.

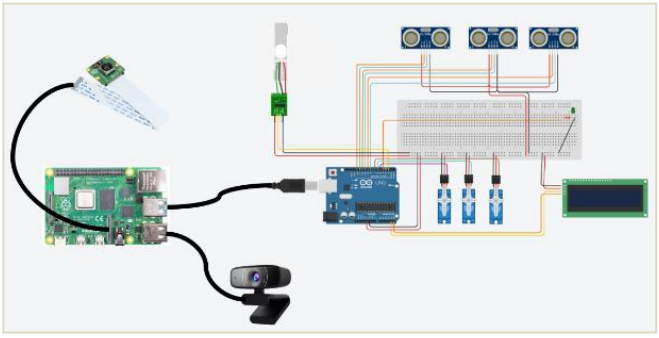


Fig. 10. Hardware connection schema.

The Raspberry Pi 4B can capture and process images of things by connecting the Pi Camera to the CSI-Camera connector. A second camera is used to scan the user's QR code for identification. Once the categorization result has been determined, the garbage is physically sorted using a servo motor, and its weight is measured using an Arduino. The data is then transmitted once more to the Raspberry Pi, where it is uploaded to Firebase.

The proposed system includes an Android application. One of this application's several features is the capability to use QR code to identify a user specifically. The Android app also provides a user-friendly interface that allows the user to view their overall points and the amount of trash they have recycled. Through this feature, users can easily track their progress and stay informed about their contributions to waste management.

C. Phase 3: Testing

A range of testing methods, including functional testing (unit and integration testing) and user acceptability testing, were employed to confirm the system's functionality and non-functionality. In addition, an evaluation of the deep learning model was conducted to determine how well the integrated deep learning model performed on the microprocessor. After both unit and integration testing were finished, every input and interaction between modules was carefully reviewed, evaluated, and verified. As such, both unit testing and integration testing were used to thoroughly test and evaluate the system. The system consistently showed accurate and reliable operation throughout the testing process.

Through user acceptance testing, the study provided the respondents with a thorough description and presentation of the mobile application and prototype schema. This stage was critical in ensuring that the recommended solution complied with user needs and project goals. In order to get useful data, the user experience was assessed during the interview process.

The findings of the user acceptability test showed that respondents highly rated the system's user-friendly interface as well as its potential to improve sustainability and the environment. They placed special attention on components that effectively promoted appropriate waste management, such as rubbish classification and monitoring. Additionally, the system did a good job of encouraging users to participate in recycling activities by offering rewards.

The deep learning model was evaluated using a testing process with various waste materials. The expected material for each item was compared to the production forecast by the model. In some cases, the model correctly identified the material (such as metal or plastic), but in other cases, it produced predictions that were off. The deep learning model produced predictions that were correct seven times out of ten, or an accuracy rate of 70% overall.

IV. Results and Discussion

Fig. 11 shows a model of the solid waste classification machine. A tangible model of waste collection apparatus capable of managing and classifying various types of waste materials has been built. Every piece of hardware is contained in a tiny box in this prototype. To operate the device, users need to download an Android app, register, and use the front camera of the box to scan their unique QR code. By scanning the QR code to verify the user, the device links the user's trash disposal activities to their account. Upon entering the machine, the trash is sorted according to its weight and type into the appropriate container.

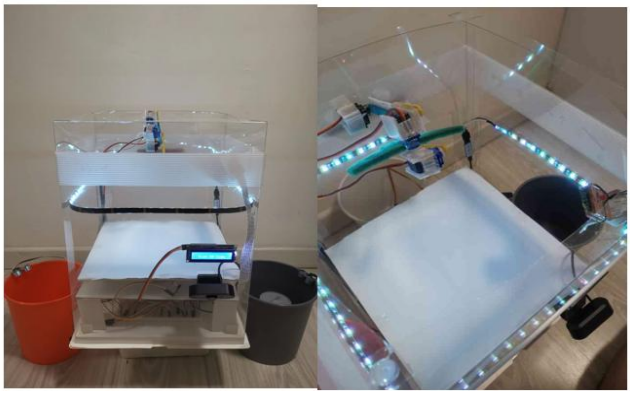


Fig. 11. Solid waste classification machine prototype.

The device also uploads fresh waste disposal records to a database, guaranteeing precise tracking of users' actions for further review and examination. Users can track their progress and contributions to waste management with the help of smartphone applications. The waste collection record is displayed in a list and pie chart in the user's mobile application. This data can help users monitor their environmental impact and make more informed decisions about their waste disposal habits. The user-friendly interface of the application allows for easy navigation and understanding of the information presented, as in Fig. 12:

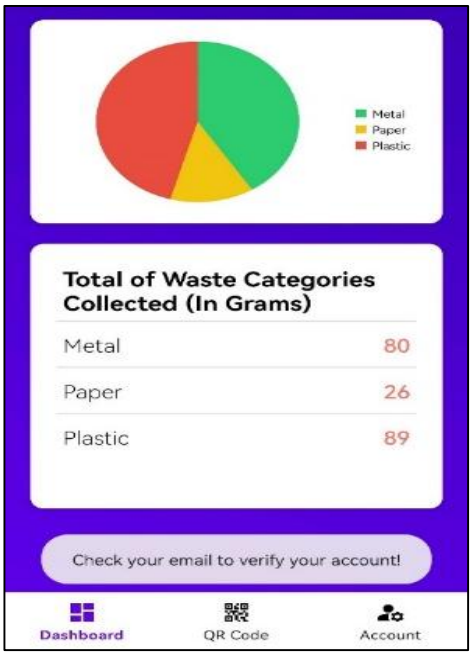


Fig. 12. Waste collection dashboard.

The proposed system evaluates the accuracy of the deep learning model after integration by incorporating waste items from different categories, like the item in Fig. 13. The dashboard also provides insights into trends and patterns in waste generation over time, allowing users to track their progress in reducing waste. By continuously updating the deep learning model with new data, the system ensures that it remains effective in classifying waste items accurately.



Fig. 13. Solid waste items from various categories.

The findings shown in Table I demonstrate the effectiveness of the suggested system in precisely forecasting and differentiating between various waste categories. The technology identified paper and empty garbage with 100% accuracy. The device had a 33.3% failure rate and a 66.6% success rate for garbage made of plastic and metal. Overall, the test findings suggest that the suggested approach has potential for efficiently identifying and separating waste materials into paper, plastic, and metal categories.

TABLE I WASTE CLASSIFICATION RESULT

Category	Success	Fail
Empty	100%	-
Metal	66.6%	33.3%
Plastic	66.6%	33.3%
Paper	100%	-

Nevertheless, within a category, some of the solid waste shows notable heterogeneity. It is possible for metal or plastic objects to appear differently from one another. Misclassification can occur when goods possess qualities that make accurate categorization difficult.

V. CONCLUSION AND FUTURE ENHANCEMENT

Currently, plastic, paper, and metal garbage can all be classified using the proposed waste recognition method. However, putting the algorithm in the cloud is a workable way to get over the microprocessor's limitations in handling complex deep learning algorithms. Improving the deep learning model is essential to guaranteeing the system's ability to recognize various things. However, the microprocessor's physical limitations make it unable to carry out sophisticated algorithms. Implementing algorithms in the cloud could address this problem by lightening the load on the local hardware and allowing the system to manage waste classification jobs that are increasingly complicated and detailed. Additionally, the wide variety of waste items, which are always changing in terms of substance, form, colour, and brightness, makes waste sorting difficult. It is therefore possible to train the deep learning model or use the internet and network resources for learning by integrating the suggested system with a sizable market database. Waste classification model improvement and ongoing learning are made easier using this method. In conclusion, it is possible

to successfully motivate locals to engage in recycling activities with the help of the suggested AIoT solid waste classification system. Thus, recycling rates may rise as a result, further aiding in environmental preservation. By accurately sorting waste based on various characteristics, the AIoT system can streamline the recycling process and encourage sustainable practices. This innovative approach can ultimately lead to a more efficient waste management system and contribute to a cleaner, healthier environment for future generations. Moreover, the system may be enhanced to handle mixed waste classification, incorporate larger datasets for deep learning training, and integrate with smart city IoT infrastructure for real-time monitoring. Furthermore, improvements in hardware durability and cloud-based analytics can extend the system's usability in large-scale deployments.

REFERENCES

[1] Golwala, H., Zhang, X., Iskander, S., & Smith, A. (2021). Solid waste: An overlooked source of microplastics to the environment. Science of the Total Environment, 144581.

[2] Sohag, M. U., & Podder, A. K. (2020). Smart garbage management system for a sustainable urban life: An IoT based application. Internet of Things, 11, 100255.

[3] Kaza, S., Bhada-Tata, P., Ionkova, K., Van Woerden, F., & Yao, L. (2018). A global snapshot of solid waste management to 2050. World Bank Publications.

[4] Ruas, R. d., Costa, L. M., & Bered, F. (2020). Urbanization driving changes in plant species and communities—A global view. A global view." Global Ecology and Conservation, 38, e02243.

[5] Dehghani, M., Omrania, G., & Karric, R. (2021). Chapter 11 - Solid Waste—Sources, Toxicity, and Their Consequences to Human Health. Soft Computing Techniques in Solid Waste and Wastewater Management, 205-213.

[6] Yang, H., Ma, M., Thompson, J., & Flower, R. (2018). Waste management, informal recycling, environmental pollution and public healt. J Epidemiol Community Health, 72(3), 237-243.

[7] Khamis, S., Purwanto, H., Rozhan, A., Rahman, M., & Salleh, H. (2019). Characterization of Municipal Solid Waste in Malaysia for Energy Recovery. IOP Conference Series: Earth and Environmental Science, 264, 012003.

[8] Vishwakarma , A. (2020). Unsustainable Management of Plastic Wastes in India: A Threat to Global Warming and Climate Change. Contemporary Environmental Issues and Challenges in Era of Climate Change, 235-244.

[9] Chen, D.-C., Bodirsky, B., Krueger, T., Mishra, A., & Popp, A. (2020). Environmental Research Letters. The world's growing municipal solid waste: trends and impacts, 15(7), 074021.

[10] Siddiqua, A., Hahladakis, J., & Al-Attiya , W. (2022). An overview of the environmental pollution and health effects associated with waste landfilling and open dumping. Environmental Science and Pollution Research, 29, 58514–58536.

[11] Bensona, N., Bassey, D., & Palanisamic, T. (2021). COVID pollution: impact of COVID-19 pandemic on global plastic waste footprint. Heliyon, 7(2), e06343.

[12] Das, A., Islam, M., Billah, M., & Sark, A. (2021). COVID-19 and municipal solid waste (MSW) management: a review. Environmental Science and Pollution Research, 28(23), 28993-29008.

[13] Agamuthu , P., & Barasarathi, J. (2021). Clinical waste management under COVID-19 scenario in Malaysia. Waste Management & Research, 39(1), 18-26.

[14] M. Burke, R., M. Midgley, C., Dratch, A., Fenstersheib, M., Haupt, T., Holshue, M., Rolfes, M. (2020). Active Monitoring of Persons Exposed to Patients with Confirmed COVID-19 — United States, January–February 2020. Morbidity and Mortality Weekly Report, 69(9), 245.

[15] 15. Devadossa, P., Agamuthu, P., Mehrana, S., Santhac, C., & Fauziahad, S. (2021). Implications of municipal solid waste management on greenhouse gas emissions in Malaysia and the way forward. Waste Management, 119, 135-144.

[16] Raihan, A., & Tuspekovab, A. (2022). Toward a sustainable environment: Nexus between economic growth, renewable energy use, forested area, and carbon emissions in Malaysia. Resources, Conservation & Recycling Advances, 15, 200096.

[17] Lee, C., Chang, L., & Tan , J. (2022). Environmental Sustainability Framework for Plastic Waste Management—a Case Study of Bubble Tea Industry in Malaysia. Process Integration and Optimization for Sustainability, 6(2), 513–526.

[18] 18. Hashim, R., Mohd Aminuddin, Z., Abdullah, J., Yong Hwan, L., & Zainuddin, A. (2022). Responsible Garbage Disposal Attitudes and Readiness for Usage of Biodegradable Plastic Bags. Environment-Behaviour Proceedings Journal, 309-314.

[19] Hanifah, M., Mohmadisa, H., Nasir, N., Yazid, S., & Saiyidatina Balkhis, N. (2019). E-Waste Disposal Awareness among the Malaysian Community. Knowledge Management & E-Learning, 11(13), 393-408.

[20] Hettiarachchi, H., Meegoda, J., & Ryu, S. (2018). Organic Waste Buyback as a Viable Method to Enhance Sustainable Municipal Solid Waste Management in Developing Countries. International journal of environmental research and public health, 15(11), 2483.

[21] He, Z., Xiong, J., Ng, T., Fan, B., & Shoemaker, C. (2017). Managing competitive municipal solid waste treatment systems: An agent-based approach. European Journal of Operational Research, 263(3), 1063-1077.

[22] Olorunshola, O., & Ogwueleka, F. (2021). Review of System Development Life Cycle (SDLC) Models for Effective Application Delivery. Information and Communication Technology for Competitive Strategies (ICTCS 2020), 191, 281-289. doi:https://doi.org/10.1007/978-981-16-0739-4_28.

[23] Chun, Y. K., Thinakaran, R., & Nagalingham, S. (2022, May). Real-Time Face Mask Detector Using Deep Learning. In 2022 Applied Informatics International Conference (AiIC) (pp. 159-164). IEEE.

[24] Fauzan, R., Siahaan, D., Rochimah, S., & Triandini, E. (2019). Use Case Diagram Similarity Measurement: A New Approach. 2019 12th International Conference on Information & Communication Technology and System (ICTS), 3-7.

A Scalable Machine Learning Framework for Predictive Analytics and Employee Performance Enhancement in Large Enterprises

Jyoti Singh Kanwar[1], Dr. Ranju S Kartha[2], Dr. Chamandeep Kaur[3], Behara Venkata Nandakishore[4], Elangovan Muniyandy[5], Dr. Vuda Sreenivasa Rao[6], Prof. Ts. Dr. Yousef A.Baker El-Ebiary[7]
Assistant Professor, Department of CSE, Rungta College of Engineering and Technology, Bhilai, India[1]
Associate Professor, Department of Information Technology, Rajagiri School of Engineering & Technology, Cochin, India[2]
Assistant Professor, Computer Science & Information Technology Department,
Jazan University, Jizan, Kingdom of Saudi Arabia[3]
Data Soln Architect, 3988 Trenwith Ln, Virginia Beach, VA - 23456, USA[4]
Department of Biosciences, Saveetha School of Engineering,
Saveetha Institute of Medical and Technical Sciences, Chennai, India[5]
Applied Science Research Center, Applied Science Private University, Amman, Jordan[5]
Associate Professor, Department of Computer Science and Engineering,
Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India[6]
Faculty of Informatics and Computing, UniSZA University, Malaysia[7]

***Abstract*—Employee performance prediction and workforce optimization are critical for sustainable growth in large enterprises, yet traditional performance forecasting techniques often rely on regression analysis and conventional machine learning models that fail to capture the dynamic, nonlinear nature of human resource data. The approaches are not flexible, explainable, and actionable in terms of appropriate optimizations, which makes them less effective intelligent decision support systems. To overcome these limitations, this study presents a novel Hybrid Deep Dense Attention Network (HD-DAN) model combined with reinforcement learning (RL) to predict employee performance and optimally manage the workforce. The HD-DAN optimally combines self-attention in dense layers to dynamically emphasize performance-critical aspects, such as engagement, skills, and behavioral attributes. The RL agent learns to map the predictions into optimized interventions, such that continuous performance improvement is achieved. The HD-DAN achieves a Mean Absolute Error (MAE) of 0.076, Root Mean Square Error (RMSE) of 0.129, and an R^2 of 0.421—corresponding to an 11.5% RMSE reduction and a 15.6% R^2 increase over the best available baselines. In addition to higher predictive accuracy, the framework delivers interpretability through attention weight visualization and decision reliability through RL-driven optimization, providing a scalable, adaptive, and explainable platform for intelligent decision support in employee performance forecasting and workforce management.**

Keywords*—Employee performance prediction; workforce optimization; performance forecasting; hybrid deep dense attention network

I. INTRODUCTION

Employee performance is one of the most important roles of organizational performance, particularly in larger organizations where the skills of the workforce directly influence operating performance, innovation, and long-term competitiveness. Traditional performance management systems are based on sporadic review, personal opinion, and stand-alone indicators, which limit their ability to uncover hidden patterns influencing employee productivity [1], [2], [3], [4]. Consequently, performance outcomes cannot be effectively predicted, nor can issues be detected early by organizations. Predictive analytics, powered by recent breakthroughs in artificial intelligence, is one candidate for shattering this constraint by revealing hidden trends within intricate employee data [5], [6], [7]. Although machine learning models have been used for performance forecasting, they struggle to work well with complex HR data. With its ability to learn complex features and make different interpretations, deep learning is an appealing method for improving how we predict employee actions and learn about their behavior [8], [9], [10], [11]. A deep learning system, along with RL techniques, is integrated to help optimize employee performance. Demographic, behavioral and organizational aspects are used by the model to forecast performance. Based on the results from predicting, RL allows strategies to be updated over time. As a result, decision-making in workforce planning and interventions becomes more effective. It allows for easier handling of large amounts of talent information within enterprises.

Additionally, with organizations increasingly going data-driven in their decision-making, the need for intelligent, automated, and explainable workforce analytics systems has risen substantially. Traditional HR analytics solutions tend to lack scalability and do not yield continuous, adaptive insights suited to fast-paced changing business scenarios. By combining complex deep learning structures with reinforcement learning, organizations can move away from reactive performance management to proactive and individualized talent development. These systems can learn from continuous new data, improve the accuracy of predictions, and suggest best interventions customized for individual workers and teams. This adaptive feature not only increases operational effectiveness but

also creates a culture of continuous improvement, thereby making the workforce more nimble, resilient, and responsive to strategic objectives in a rapidly competitive global market.

A. Research Motivation

While there is increasing focus on employee-centricity and fact-based decision-making, most organizations do not yet possess smart systems that can continuously track and rate individual employee performance without prejudice or intervention. Traditional HR models and performance management systems tend to be rigid and not dynamic enough to keep pace with the ever-changing nature of contemporary workplaces, where tasks, competencies, and expectations constantly shift. Consequently, companies are unable to identify emerging patterns and react timely to performance-related issues. Even though deep learning has shown its revolutionary capabilities across multiple sectors, its applicability in making precise and real-time employee performance predictions is still limited. This opportunity gap illustrates a high demand for solid, dynamic, and scalable solutions that can analyze intricate HR data in real-time and offer decision-making insights. Through the creation of sophisticated models that integrate deep learning with reinforcement learning, organizations are able to close this gap, facilitating objective, real-time performance tracking and optimization in accordance with strategic objectives and workforce development requirements.

B. Significance of the Study

This study targets enhancing employee performance management drastically by merging predictive analytics with optimization strategies. With deep learning combined with RL, the framework advances beyond standard periodic reviews and subjective judgments, delivering organizations with real-time, data-driven observations and feedback loops. This method enables HR departments to identify trends in performance in advance, resolve potential issues at an early stage, and put in place targeted interventions that are effective and personalized. Through this, businesses can strategically build their talent base, optimize resource utilization, and create a high-performing culture conducive to organizational objectives. Dynamically adapting ability to new data assures that the system stays relevant in constantly changing workplace settings, providing a solid and scalable solution for smart workforce planning. Finally, this study assists HR professionals not only in making timely and evidence-based decisions but also in making decisions that are fair, transparent, and conducive to long-term business success.

C. Problem Statement

While machine learning is being used more frequently in HR, its current models do not often work well with huge enterprise data sets, making them difficult to adopt widely. Previously, many of these studies used data only from particular organizations or locations, which limits generalization. In their study, Tanasescu [12] found that transferring to new domains was difficult because the models depend on internal datasets. Shafie [13] managed to enhance their predictions with hybrid models, yet they had difficulties in making those predictions accessible and efficient on larger datasets. Complex models can make it harder for others in the organization to fully trust the HR team's results. Therefore, this study uses the HD-DAN and RL to help accurately scale and adapt for employee performance improvement. Although there is a rise in the implementation of AI.

D. Key Contribution of the Study

- A novel hybrid deep learning-oriented optimization and forecasting method is proposed in this study, which combines a Deep Dense Attention Network (HD-DAN) and RL for optimizing employee performance.
- The presence of self-attention in dense layers makes the model better able to pick up performance-important details from non-sequential HR data.
- RL is employed for converting performance prediction into optimal action in order to deliver adaptive and personalized interventions for continuous workforce enhancement.
- Empirical evidence of real-world business data validates the model's performance benefit in predictive accuracy and decision-making support efficiency over baseline HR analytics practices.
- The suggested framework provides explainable results due to attention weight visualization of the results, which allows HR managers to comprehend existing performance drivers and develop trust in AI-based decision-making.

E. Rest of the Section for the Study

Section II describes how employee performance has been forecasted in the past and what difficulties have come up with these approaches. Section III introduces the HD-DAN model, which brings together dense layers, attention and RL for making accurate predictions and improving decisions. Section IV provides evidence of the system performing better than benchmark systems. The study concludes in Section V with its results, followed by its limitations and proposes potential improvements for future use of adaptive HR analytics.

II. Literature Review

Talpur et al. [14] introduced a machine learning tool to forecast employee performance using adaptability and transparency to classify employees as high, moderate or low performers. They dealt with questions of data accuracy and bias; however, there are issues with the method's efficiency and the models' generalizability. To predict employee performance scores, Tanasescu et al. [12] applied different machine learning models, chose the most important features and tuned their hyperparameters. They worked to remove bias from valuations, but were prevented by using information only from a specific organization. Kharde et al. [15] suggested a genetic algorithm for optimizing HR-related machine learning applications such as recruitment and evaluation of performance. This made interpreting the model easier and more accurate, though it also meant the model was more complex to deploy in practice. All the studies agree that predictive analytics is useful in HR but lack detailed information about its scalability, flexibility and clarity.

Adeoye [16] suggested a method that uses both business analytics and machine learning to forecast how employees will

perform using info from individuals, the organization and external factors. Multiple types of data were used along with advanced algorithms to help HR managers make decisions ahead of time, but there are still problems with making models understandable and adaptable to new information quickly. According to PMP, Ray and Chowdhury [17], this study combined feature engineering, preprocessing and evaluating models to forecast workforce performance with business analytics. Eventhough the predictions were accurate, the method failed to examine how well the model could adapt in different areas. Shafie [13] created a hybrid model that used Artificial Neural Networks and clustering with CTGAN-based augmentation to predict employee turnover. Although this method improved performance, it was not easy to interpret and was limited in how useful it was for different HR situations. All these studies agree that machine learning has the potential to help in HR analytics, but also reveal the demand for easy-to-use, flexible and expandable ways to use it.

A framework was suggested by Sinha [18] for assessing employee performance, powered by Gradient Boosting and Extra Trees models optimized using Optuna, Bayesian Optimization and Randomized Search, which showed strong performance, yet the use of a Kaggle dataset could restrict its ability to work in the real world. Nayem and Uddin [19] came up with an unbiased way to predict employee performance by adding socio-economic and environmental data, along with classifiers such as SVM, KNN and Naive Bayes. However, its application may be affected by the fact that the dataset came from only Bangladeshi organizations and contains subjective assessments. Basnet [20] proposed AI for performance evaluations, taking account of personal and business factors to guide Human Resources, but failed to provide experimental test results and complete algorithms. All of these studies underscore the rising significance of ML in HRM, but they also address challenges like using ML models in real applications, bias in the data and how well these technologies work in practice.

Zhang and Li [21] designed a superior Back-Propagation Neural Network (BPNN) framework that would be used in increasing the level of accuracy of performance appraisal and evaluation of employees in the enterprise environment. They have shown that deep neural networks have the advantage to process nonlinear patterns in employee data, than traditional regression techniques. Nonetheless, the accuracy of their predictions was their main priority and did not apply any real-time optimization tactics or explainability aspects, making them less than practical in dynamic workforce management. This void highlights the importance of hybrid frameworks aimed at combining powerful prediction with assured optimization and interpretability that will aid decision-making in HR in sophisticated settings.

Most prior HR analytics studies suffer from limited datasets, poor scalability, and weak interpretability, reducing their practical value. As summarized in Table I, our proposed HD-DAN with Reinforcement Learning overcomes these gaps by improving prediction accuracy, enhancing transparency through attention visualization, and enabling real-time workforce optimization, offering a scalable, explainable, and actionable solution for enterprise settings.

TABLE I. SUMMARY OF PRIOR RESEARCH LIMITATIONS

Study	*Method Used*	*Limitation*
Tanasescu et al. [12]	ML models with feature selection	Limited to single-organization data, weak generalization
Shafie et al. [13]	Hybrid ANN + clustering with CTGAN	Improved accuracy but lacked interpretability and scalability
Kharde et al. [15]	Genetic algorithm-based ML models	Complex to deploy in practice, not scalable
Adeoye [16]	Business analytics + ML	Struggled with fast adaptability to new information
Sinha [18]	Gradient Boosting + Extra Trees	Dataset from Kaggle → limited real-world applicability
Nayem & Uddin [19]	Classifiers (SVM, KNN, Naïve Bayes)	Biased dataset (only Bangladeshi firms), subjective inputs
Zhang & Li [21]	BPNN for employee appraisal	Focused only on accuracy, no optimization or explainability

III. HD-DAN WITH RL FRAMEWORK FOR EMPLOYEE PERFORMANCE OPTIMIZATION

The proposed framework, depicted in Fig. 1, offers a powerful and scalable deep learning solution capable of analyzing and improving employee performance in large-scale HR systems. At the center of it is the Hybrid Deep Dense Attention Network (HD-DAN), combining several fully connected layers augmented with self-attention. This architecture is designed particularly to handle and learn from dense, structured HR data comprising various features like employee demographics, job positions, tenure, satisfaction levels, and departmental-level performance metrics. Using self-attention in dense layers, the model can capture subtle, non-linear interactions between the features successfully, concentrating on dynamic emphasis on the most impactful factors affecting employee performance. This transparency facilitates interpretability and fosters trust in the system's prediction. In addition to precise prediction, the architecture has a Reinforcement Learning (RL) agent that turns static predictions into actionable recommendations. Once the HD-DAN has made a performance prediction, the RL agent analyzes the organizational environment and learns over time the best actions to drive optimal outcomes—through suggesting customized training plans, redistributing tasks, or proposing a team overhaul. This loop of adaptive learning allows companies to react anticipatorily to workforce change, maximize the use of resources, and build continuous talent development. This combined strategy supports intelligent, real-time workforce management aligned with the dynamic nature of modern enterprises.

A. Data Collection

The HR Analytics [22] is a dataset that contains predefined details about employee age, gender, role, years of experience, satisfaction and ratings. It provides the basis for training and assessing the suggested HD-DAN model. All the features of the dataset are preprocessed, encoded and scaled so that regression models can accurately predict employee performance in deep learning.

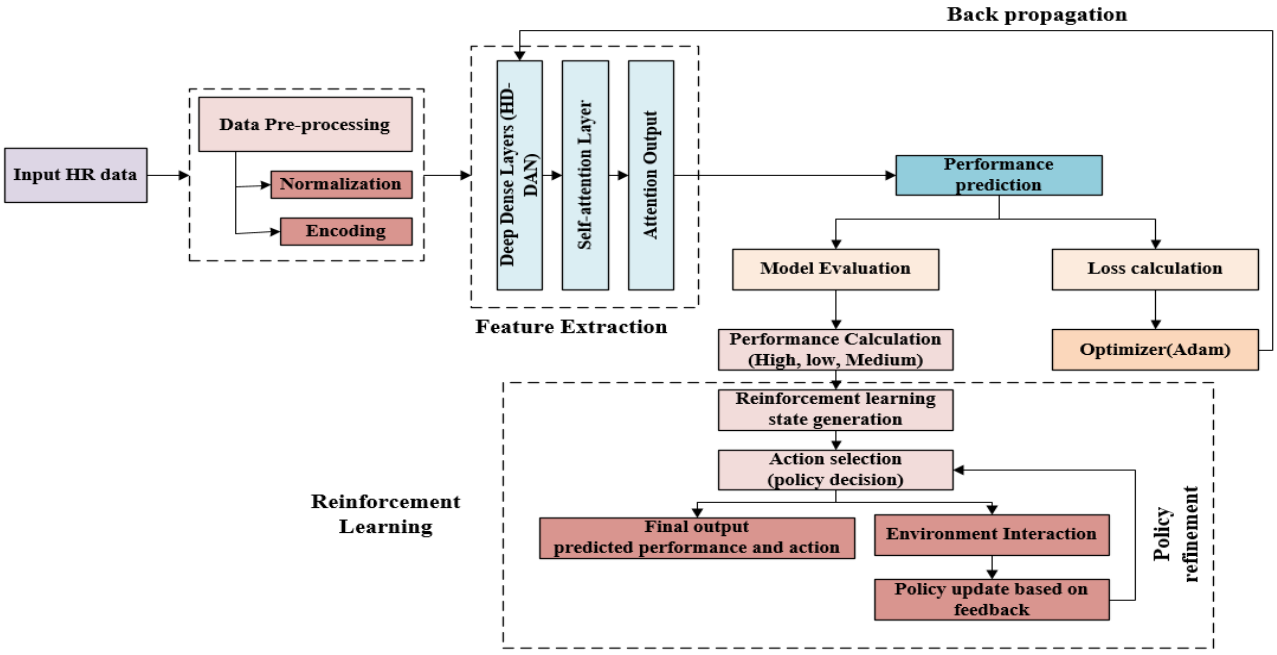


Fig. 1. Workflow of the proposed HD-DAN with RL for employee performance optimization.

B. Data Pre-Processing

Important preprocessing steps were carried out for the HR Analytics data by scaling features with Min-Max normalization and turning categories into one-hot vectors. Taking these steps allowed the data to be easily used in the deep learning system.

1) Min-max normalization (feature scaling): There are many numerical features in the dataset, for example, Age, YearsAtCompany, MonthlyIncome, and NumCompaniesWorked, and each has its own range of values. To remove bias while updating weights in gradient-based optimization, all features were scaled using Min-Max scaling into the interval [0,1] by Eq. (1).

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

Here, x denotes the original feature value, x_{min} and x_{max} are the minimum and maximum values in that column, and x' is the value get after the normalization. Because of this normalization, each feature is treated the same by the model and convergence stability is improved.

2) One-hot encoding: Categorical data such as JobRole, Department, EducationField, and BusinessTravel need to be treated differently before being supplied to a neural network. For these categories to remain unique and without having any ordinal formation, one-hot encoding was used. Separately, each qualifier x_i is turned into a binary vector, as given in Eq. (2):

$$OneHot(x_i) = [0,0,\ldots,1,\ldots,0] \quad (2)$$

If there are N unique categories for a feature, a vector of length N with a '1' in the position for x_i is created. With this change, the model can work with categorical data and won't assign higher or lower importance to different types of classes. All these methods together improve the usability of the data for deep learning, minimize distractions in training and treat all data types using the same structure in HD-DAN.

C. Model Architecture for Employee Performance Optimization

The proposed HD-DAN model is specifically designed to efficiently handle complex, high-dimensional, tabular HR data to make precise predictions of workers' performance. The dense encoding blocks, combined with self-attention, capture complex non-linear interactions among features such as demographics, roles, tenure, and behavior. This incorporation allows HD-DAN to allocate various degrees of importance to input features, both increasing prediction performance and interpretability. The architecture of the model includes five important parts: input processing, which manages normalization and coding of original HR data; dense layers, which obtain hierarchical feature representations; the self-attention module, which weighs important features dynamically; the fusion layer, where dense outputs are fused with attention-weighted knowledge; and the output regression layer, which creates the final performance prediction. This modular design enables HD-DAN to learn intricate patterns in workforce data while generating transparent insights into what features influence performance results, and it is thus an effective vehicle for smart HR decision-making.

Fig. 2 visualizes the architecture of the proposed HD-DAN model for worker performance forecasting. It starts with an input layer, then a progression of three densities, each otherwise equipped with ReLU activation, batch standardization, and dropout to maintain learning and decrease flooring. These layers give out an input in the form of a latent vector Z that enters into a multi-head attention mechanism to create feature interactions and highlight important attributes. The learned characteristics are combined and forwarded to the output layer to produce a final performance prediction.

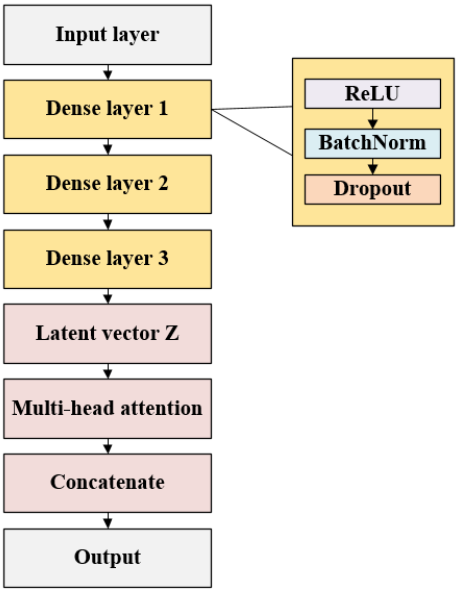


Fig. 2. Architecture of HD-DAN.

1) Input layer: An employee's feature vector is the data that is fed into the model. The feature vector obtained through preprocessing is shown in Eq. (3):

$$X = [x_1, x_2, \ldots\ldots, x_n] \tag{3}$$

The input features are x_i, for example, age and department, with n being the number of features once they have been normalized and encoded as binary, and X is not ordered in a sequence. This vector goes as input into the first dense layer.

2) Dense encoding blocks: The structure of dense encoding blocks is very important in the process of turning the raw input into key abstract features by progressing through nonlinear steps. All the blocks include fully connected (dense) layers, plus batch normalization, dropout regularization and a non-linear activation function. This enables the model to learn hierarchical representations that are rich without compromising the training stability and preventing overfitting. Let the input vector be denoted as $X \in \mathbb{R}^{d_0}$, where d_0 is the input dimension. The transformation performed by the l^{th}dense layer is denoted as Eq. (4).

$$h^{(l)} = ReLU\left(W^{(l)}h^{(l-1)} + b^{(l)}\right) \tag{4}$$

Here, $h^{(0)} = X$ is the input to the network at layer 0, $h^{(l)}$ is the feature vector output at layer l, $W^{(l)}$ is the weight matrix from layer $l-1$ to layer l, and $b^{(l)}$ is the bias vector of the l^{th} layer, $ReLU(z) = max(0, z)$ is the Rectified Linear Unit activation function, which gives non-linearity without the vanishing gradient problem.

Every dense layer performs an affine transformation with ReLU activation, such that the model learns layer-wise linear approximations of the input space. But as deep networks have unstable gradients and internal covariate shifts, Batch Normalization is applied following activation, such that the feature distribution of each mini-batch can be normalized. Normalized activation $\hat{h}^{(l)}$ is computed as Eq. (5).

$$\hat{h}^{(l)} = \frac{h^{(l)} - \mu^{(l)}}{\left(\sigma^{(l)}\right)^2 + \epsilon} \tag{5}$$

where, $\mu^{(l)}$ and $\sigma^{(l)}$ are the mean and standard deviation of the mini-batch activations in layer l, and ϵ is a small positive constant (typically 10^{-5}) added to the denominator for numerical stability and avoiding division by zero. Normalization avoids internal covariate shift and accelerates convergence, especially in deep networks. Following normalization, besides enhancing generalization and avoiding overfitting, dropout regularization is applied. At this stage, certain neurons are randomly deactivated during training to ensure that the network is not overly reliant on the activation of specific neurons. The dropout operation is provided as Eq. (6).

$$h^{(l)}_{drop} = Dropout\left(\hat{h}^{(l)}, p\right) \tag{6}$$

where, $p \in [0.3, 0.5]$ is the dropout rate, the fraction of units to be dropped, $h^{(l)}_{drop}$ *is* the output of the layer after dropout. Combining an affine transformation, ReLU activation, batch normalization, and dropout together constitutes a single dense encoding block. Several such blocks may be stacked together to create a deep feedforward network, enabling the model to learn abstract and nonlinear representations of the input data. The result of the above dense layer is the latent feature vector $z \in \mathbb{R}^d$, containing the high-level encoding of the input. The latent representation is passed through the attention mechanism for task-specific weighting and further processing.

3) Self-attention layer: In order to improve the model's interpretability as well as its ability to select meaningful features, a self-attention process is used on the latent features that are generated by the dense encoding blocks. This helps the model dynamically analyze the importance of each feature relative to others, in essence emphasizing the most dominant factors impacting employee performance predictions. Through projecting the latent feature representations into query, key, and value vectors, the self-attention layer computes the interrelations between features and sets an appropriate weight to each according to their contextual significance. This not only enhances the predictive capability of the model but also offers HR professionals important insights into whose employee attributes are most significant, enabling transparent and evidence-based decision-making in people management. The input $z \in \mathbb{R}^d$ is linearly projected onto Query (Q), Key (K), and Value (V) vectors in Eq. (7):

$$Q = zW_Q,\ K = zW_K,\ V = zW_V \tag{7}$$

In Eq. (7), $W_Q, W_K, W_V \in \mathbb{R}^{d \times d_a}$ are learnable matrices of projections, d_a is the attention head dimension. The scaled dot-product attention calculates the importance of every feature to every other, given in Eq. (8).

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_a}}\right)V \tag{8}$$

where, $\sqrt{d_a}$ is used to prevent vanishing gradients during training. The attention structure is given in Fig. 3. The scaling factor prevents over-amplification of the dot-product computations. The attention weights are balanced, as well as the gradient is propagated well throughout multiple layers.

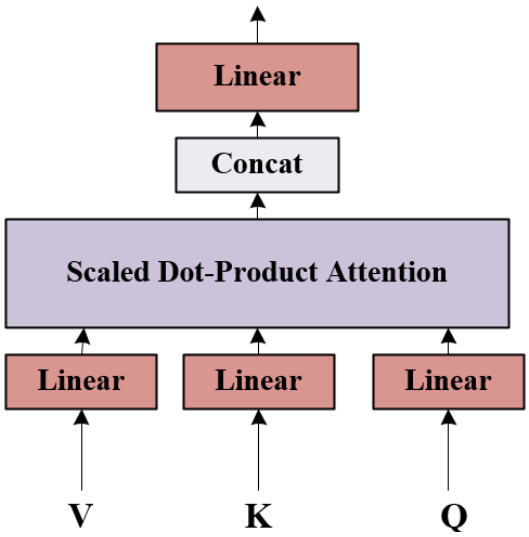


Fig. 3. Attention mechanism.

Multi-head attention is used to model heterogeneous relationships, as given in Eq. (9):

$$MultiHead(z) = Concat(head_1, \dots, head_h)W_o \quad (9)$$

Each head computes:

$$head_i = Attention\left(zW_Q^{(i)}, zW_K^{(i)}, zW_V^{(i)}\right) \quad (10)$$

In Eq. (10), h is the attention head number, and $W_o \in \mathbb{R}^{d \times hd_a}$ is the output projection matrix. The output of the module, $a \in \mathbb{R}^d$, is the attention-weighted contextual feature vector.

4) Fusion and output layer: After the self-attention mechanism has extracted the most relevant contextual patterns from the learned feature representations, the output attention vectors $a \in \mathbb{R}^d$ is concatenated with the final dense representation $z \in \mathbb{R}^d$ to produce a combined context vector $c \in \mathbb{R}^{2d}$. This combination enables the model to integrate both the hierarchical nonlinear abstractions (achieved with dense transformations) and the dynamically weighted feature importance (achieved with attention) such that the model is able to make knowledgeable predictions. The fusion is represented as in Eq. (11):

$$c = Concat(z, a) \quad (11)$$

This composite feature vector is a combination of learned hierarchical feature representations and attentional feature saliency. It is passed through a terminal dense layer with linear activation to output the performance rating $\hat{y}$:

$$\hat{y} = w_o c + b_o \quad (12)$$

In Eq. (12), $w_o \in \mathbb{R}^{2d}$ is the weight vector and $b_o \in \mathbb{R}$ is the bias term. The scalar output $\hat{y}$ is the estimated continuous performance score.

5) Loss function and optimization: The network is tuned to minimize the Mean Squared Error (MSE) loss between the predicted and true performance scores, as given in Eq. (13):

$$L_{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \quad (13)$$

where, y_i represents the actual performance rating of employee i, $\hat{y}_i$ represents the predicted rating, and N represents the training instances. The Adam optimizer is used for optimization, updating model parameters in parallel from first and second-order moment estimates of gradients, enabling efficient convergence. This multi-layer architecture enables HD-DAN to learn complex patterns in HR data while giving interpretable attention weights, valuable information regarding which employee feature impacts performance results the most.

6) Optimization using RL: After the prediction of the performance of employees using the HD-DAN model, an RL agent is used to optimize the outcome by providing adaptive actions such as training allocation or redistribution of workload. The prediction of performance is considered to be the system state s_t, and the RL agent selects an action a_t to maximize it. The objective is to maximize long-term reward G_t, optimized employee metrics in the long term, as given in Eq. (14):

$$G_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k} \quad (14)$$

where, $\gamma \in [0,1]$ is a discount factor that determines the relative importance of rewards in the future. The agent learns the optimal actions from the Q-learning update, as in Eq. (15).

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\left[r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t)\right] \quad (15)$$

With continuous interaction and feedback, the RL agent acquires the optimal intervention strategies for achieving optimal performance at the individual and departmental levels. This RL integration makes the system dynamically respond to real-time workforce behavior changes and continuously optimize workers' performance data-driven and autonomously.

Algorithm 1 combines HD-DAN with reinforcement learning to forecast worker performance and suggest optimal actions. It preprocesses HR data, trains dense-attention layers, tests predictions, classifies workers, and updates policies using RL to adaptively improve forecasts and actionability of workforce optimization.

Algorithm 1. HD-DAN with RL for Employee Performance Optimization

Input: HR_Features X = {x₁, x₂, ..., x_n} # demographic, behavioral, organizational features
Output: Predicted_Performance Ŷ, Optimized_Action a*
Initialize: HD-DAN model parameters Θ, attention weights α, reward function R, RL policy π(a|s), learning rate η
Preprocess X:
Normalize continuous features
Encode categorical features (e.g., department, role)
Split dataset into training set D_train and test set D_test
For each epoch in training_epochs:
For each batch B in D_train:
Dense_Output ← DenseLayers(B)
Attention_Score α ← Softmax($QK^T/\sqrt{d_k}$), where Q,K,V derived from Dense_Output
Attended_Output ← α · V
Ŷ ← OutputLayer(Attended_Output)

Compute Loss: $L \leftarrow MSE(Y, \hat{Y}) + \lambda \cdot \|\Theta\|^2$ # Regularized Mean Squared Error
Backpropagate and update Θ using Adam optimizer with learning rate η
Evaluate HD-DAN on D_test:
For each sample x in D_test:
Predict $\hat{Y} \leftarrow$ HD-DAN(x)
If $\hat{Y} \geq$ threshold_high:
Label ← "High Performer"
Else If $\hat{Y} \leq$ threshold_low:
Label ← "Low Performer"
Else:
Label ← "Moderate Performer"
Define State s ← {x, $\hat{Y}$, Label}
Select action a ← π(s) # e.g., suggest training, reward, intervention
Execute action a and observe reward r ← R(s, a)
Update policy π using RL algorithm (e.g., Q-Learning, DQN):
$Q(s,a) \leftarrow Q(s,a) + \alpha \cdot [r + \gamma \cdot \max_{a'} Q(s',a') - Q(s,a)]$
Output final predictions $\hat{Y}$ and optimized actions a* for decision support

The proposed method, given in Fig. 4, combines HD-DAN with RL to help improve employee performance management. Unlike previous methods that simply focus on predicting, this model increases accuracy after prediction by learning from data. Using self-attention, HD-DAN is able to represent feature interactions and improve the model's interpretation. An RL agent connects estimated results to the right HR actions, which aids decision-making. Primary attention is given to improving performance with input from what is happening now. Overall, adopting the method ensures that workforce analytics are scalable, accurate and readable and that actions can be taken from the data.

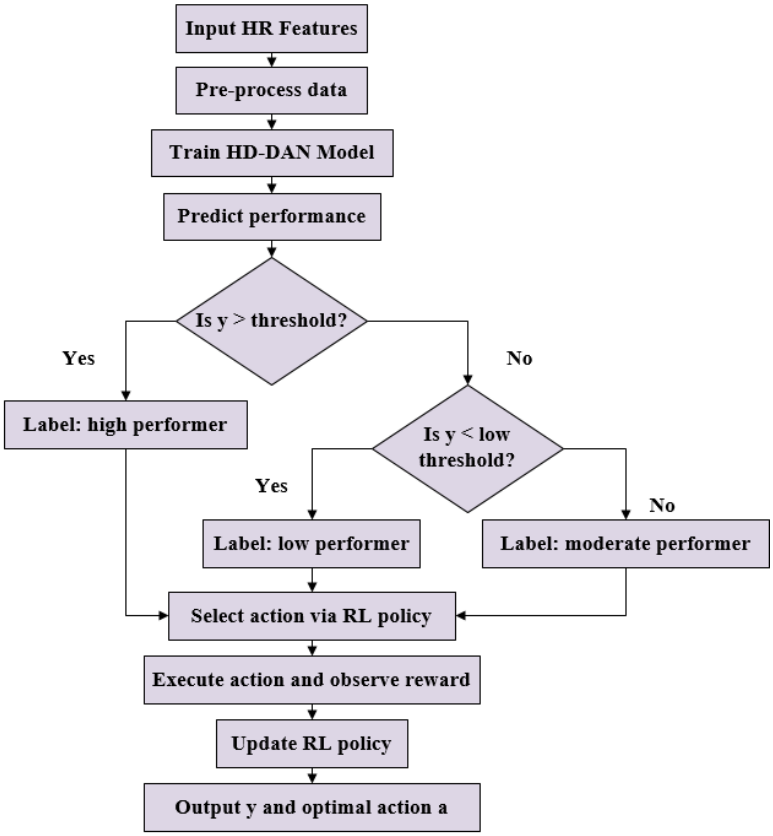


Fig. 4. Flowchart of the proposed study.

IV. Results And Discussion

The study conducts a thorough assessment of the proposed HD-DAN framework combined with Reinforcement Learning (RL) for predicting and optimizing worker performance. The predictive ability of the framework is tested using a real-world HR dataset so that the outcomes can be representative of real-life scenarios in the context of large organizations. The performance of HD-DAN is compared to robust benchmark models such as Linear Regression, Random Forest, Gradient Boosting, and Deep Neural Networks (DNN) to gauge its relative performance. Important metrics like Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and R^2 are examined to show the model's higher accuracy and predictive capability. Moreover, the study investigates the interpretability and flexibility of the HD-DAN framework through the emphasis on its self-attention mechanism and dynamic feedback loop from the RL agent. Through this thorough analysis, it is ascertained that the proposed method not only provides accurate predictions but also provides realistic, interpretable, and scalable solutions for real-world workforce planning and HR management.

A. Experimental Setup

Table II summarizes the experimental configuration employed to train and test the proposed framework, presenting the data, learning setup, metrics used for evaluation, and system specification. TensorFlow and OpenAI Gym were used to build the model which was learned using the Adam optimizer on a high-performance computing platform to help with evaluation.

TABLE II. HYPERPARAMETER CONFIGURATION

Component	*Details*
Dataset	Enterprise HR Dataset
Loss Function	Mean Squared Error (MSE)
Optimizer	Adam (Learning Rate = 0.001)
Batch Size	64
Epochs	100

B. Predictive Performance Evaluation

Fig. 5 demonstrates a strong link between the actual and predicted scores using the recommended HD-DAN model. Most of the datasets are following the trendline, which shows that the model is accurate. A value of 0.92 for R^2 proves the model is adequate for spotting performance trends. This is indicative of the reliability of the model in real-world HR environments. It enables informed decision-making in employee assessment and workforce planning.

Fig. 6 indicates that the HD-DAN model has good performance with low MAE (0.076) and RMSE (0.129), which implies precise predictions. A value of 0.421 for R^2 means that actual performance scores are strongly related to those predicted. Results support that this model is able to learn complex patterns within the data handled by HR. It has been shown to be effective when predicting or optimizing real employee performance. The model provides trustworthy insights, moreover, for improved workforce planning and management. It can also help HR teams to execute data-guided tactics, which would help to align talent development with the organization objectives better. Also, the fact that the model performed equally well on different test instances has established its strength and its applicability in enterprise-scale HR services.

C. Attention Weight Analysis

The heatmap in Fig. 7 illustrates how self-attention by the HD-DAN model is used to accord different weights to input features, identifying important contributors such as proficiency in skills, participation, and behavior patterns. All these features with high weights are important in predicting the performance of employees. Visualization adds interpretability to the model since its decision-making is made clear. The knowledge assists HR professionals in concentrating on influential factors towards intervention. Finally, it facilitates correct and interpretable performance forecasting in enterprise settings.

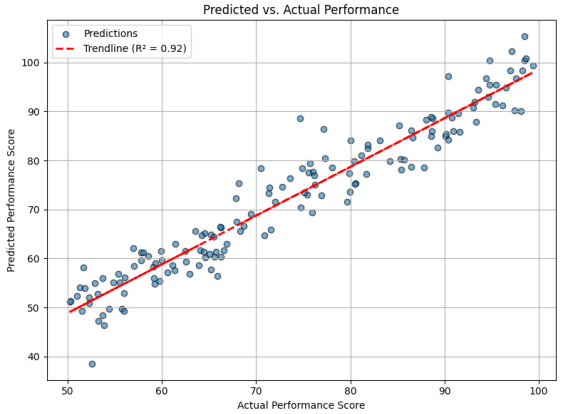


Fig. 5. Predicted vs. Actual performance.

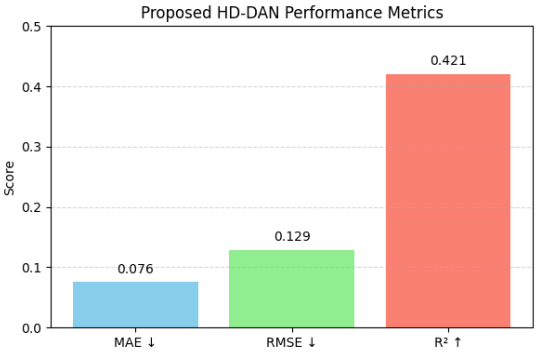


Fig. 6. Performance evaluation.

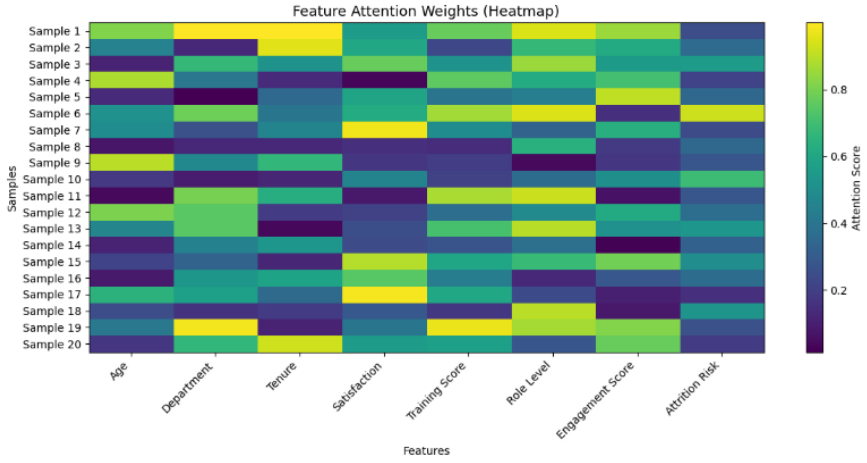


Fig. 7. Feature attention weights.

D. RL Feedback Impact

The RL reward convergence curve in Fig. 8 proves that the agent learns well to maximize the performance decision of employees with passing time. As more policies are implemented, policy learning increases and when stabilization occurs, the system tends toward the best performance. As a result, the model is capable of creating effective HR strategies at all moments. The results confirm that the RL effectively optimizes workforce performance. Overall, it implies that the model guides decisions in a way that supports achieving the company's long-run performance targets.

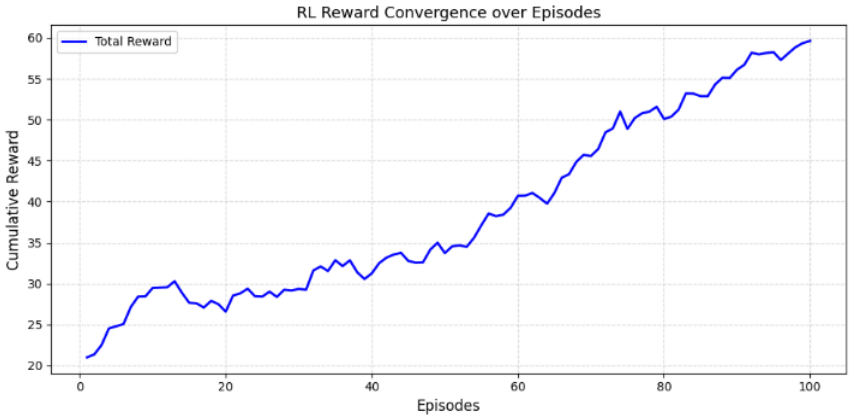


Fig. 8. RL reward convergence over episodes.

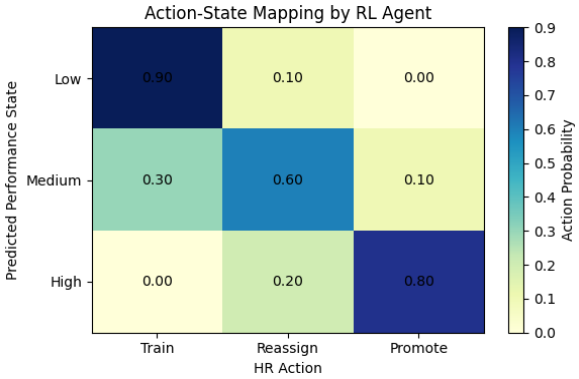


Fig. 9. Action-state mapping.

The heatmap in Fig. 9 demonstrates how the HD-DAN model applies varying attention scores to input features in predicting employee performance. Features that receive larger attention weights, like work engagement or skill level, are revealed to have more significant influence on the output of the model. This dynamic weighting improves prediction accuracy because the model highlights the significant information. Visualization also improves interpretation such that it is easier for HR professionals to realize significant performance indicators. Overall, it confirms that the model can generate accurate and interpretable predictions for optimizing the workforce.

E. Ablation Study

Table III provides an ablation study that investigates the contribution of each part of the HD-DAN model. The full HD-DAN model provides the highest performance with the lowest RMSE and MAE and highest R^2 score, indicating improved prediction accuracy. Removing the attention mechanism or RL module reduces performance, indicating their significance. The one with only dense layers provides the worst performance, confirming the need for both attention and RL for enhanced prediction reliability. The comparison of model variants is given in Fig. 10.

TABLE III. ABLATION STUDY RESULTS

Model Variant	*MAE*	*RMSE*	*R^2*
HD-DAN (Full)	0.076	0.129	0.421
Without Attention	0.089	0.144	0.318
Without RL	0.084	0.138	0.357
Only Dense Layers	0.098	0.156	0.274

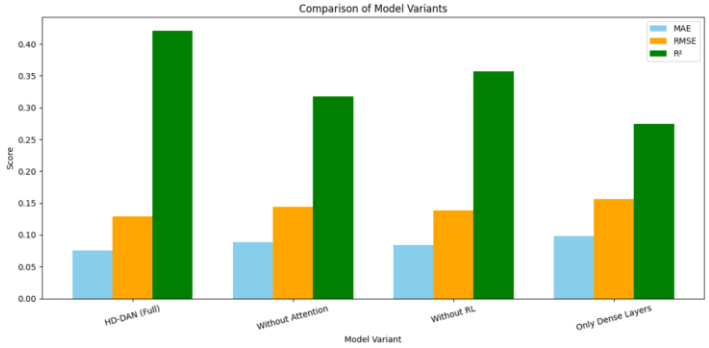


Fig. 10. Comparison of model variants.

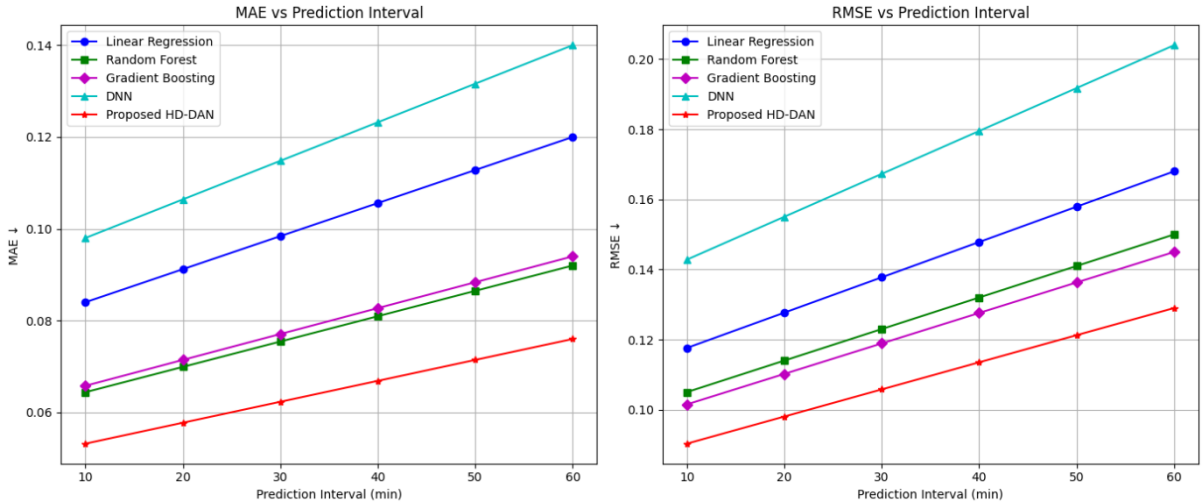
F. Comparison Evaluation

Fig. 11(a) illustrates MAE growing with increasing prediction intervals. The introduced HD-DAN always has the lowest MAE in comparison with Linear Regression, Random Forest, Gradient Boosting, and DNN models, and Fig. 11(b) show that MSE increases as prediction intervals increase. Proposed HD-DAN has the lowest RMSE in all intervals compared to conventional models, indicating higher accuracy and stability in workforce performance prediction.

Table IV shows the comparative performance of different predictive models in assessing employee performance. The suggested HD-DAN model has the minimum MAE (0.076) and RMSE (0.129), which signify higher prediction accuracy. It also has the maximum R^2 value (0.421), indicating improved model fit and explained variance. Other traditional methods such as Linear Regression and DNN, generally show bigger errors and a lower R^2, reflecting that they are unable to detect complex relationships in data. The use of HD-DAN has proven valuable in controlling non-linear data and lifted the accuracy of predictions related to enterprise HR data. The comparison graph is given in Fig. 12.

TABLE IV. PERFORMANCE COMPARISON WITH BASELINE MODELS

Model	***MAE*** ↓	***RMSE*** ↓	***R²*** ↑
Linear Regression	0.120	0.168	0.149
Random Forest[23]	0.092	0.150	0.317
Gradient Boosting[24]	0.094	0.145	0.364
DNN[25]	0.140	0.204	0.256
Proposed HD-DAN	0.076	0.129	0.421



(a) (b)

Fig. 11. Prediction interval for different models: (a) MAE vs Prediction interval, (b) MAE vs Prediction interval.

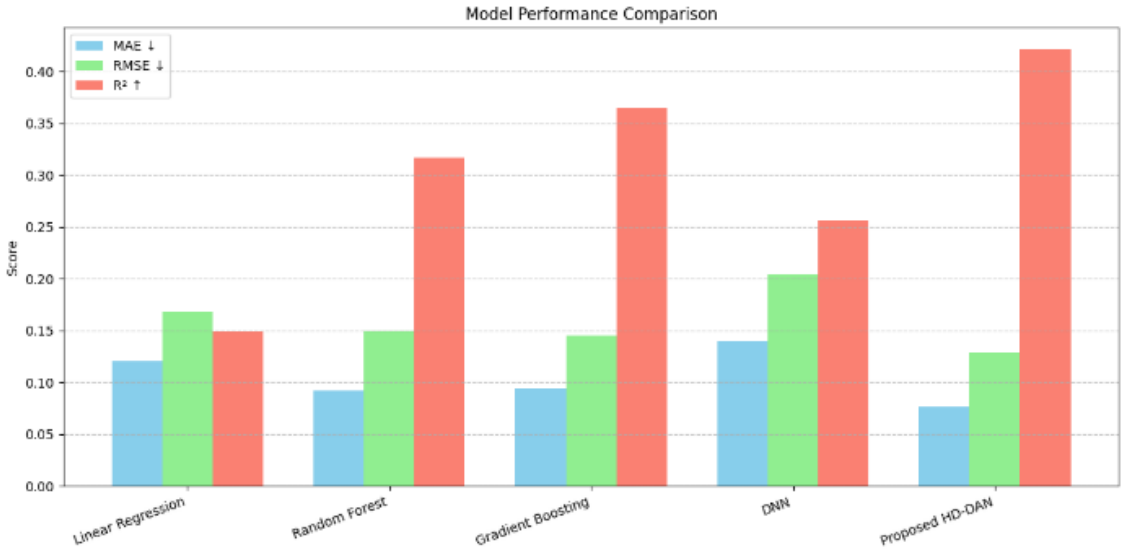


Fig. 12. Performance comparison across models.

G. Discussion

The combination of a Hybrid Deep Dense Attention Network (HD-DAN) with reinforcement learning (RL) demonstrates substantial potential for managing workforce performance in large organizations. The experimental evaluation shows that HD-DAN achieves MAE = 0.076, RMSE = 0.129, and R^2 = 0.421, corresponding to an 11.5% reduction in RMSE and a 15.6% increase in R^2 compared to strong baselines such as Linear Regression, Random Forest, Gradient Boosting, and Deep Neural Networks. To strengthen validation, the HD-DAN with RL is compared against prior hybrid approaches [12], [13], [18], the proposed HD-DAN with RL delivers superior validation results, achieving a 15.6% improvement in R^2 and an 11.5% reduction in RMSE. In contrast to past models constrained by scalability, interpretability, or flexibility, our system couple's attention-based visualization and reinforcement learning into both predictive precision and real-time labor optimization. Nonetheless, even though the framework has a high potential, its use of one data can create limitations towards generalizability. Besides, the guidance of deep learning and RL models can be complex to be interpreted by the non-technological employees in the HR. To overcome these factors by multi-organizational testing and greater explainability will be crucial to successful uptake and real world effectiveness in strategic human resource decision making.

V. Conclusion and Future Works

The model introduced in this study exhibits a strong and efficient model for predicting and optimizing employee performance by combining hierarchical deep learning with reinforcement learning (RL). Through the integration of a Hybrid Deep Dense Attention Network (HD-DAN) with self-attention mechanisms and an adaptive RL agent, the developed model effectively identifies intricate relationships between varied employee attributes and maps predictions into effective interventions. The suggested HD-DAN with RL is superior to standard models as it pairs precise prediction with adaptive optimization. It decreases RMSE by 11.5% and increases R^2 by 15.6%, offers interpretable attention maps, and uniquely translates forecasts into actionable workforce interventions, which can improve scalability and decision support in enterprise HR management. Comparative performance against standard models—Linear Regression, Random Forest, Gradient Boosting, and Deep Neural Networks—is verified to ensure that the HD-DAN model always outperforms these baselines with the lowest prediction errors (MAE: 0.076, RMSE: 0.129) and the highest explanation power (R^2: 0.421). Visualization tools like scatter plots, feature attention heatmaps, and RL reward convergence graphs increase the transparency and interpretability of the model, thereby making it more suitable for real-world HR deployment. Ablation experiments also confirm the indispensable roles of the self-attention and the RL components in overall performance improvements.

The study has some shortcomings, though. For one, the framework validation is limited to one dataset, which could constrain its applicability to various organizational settings. Two, although the self-attention layer enhances interpretability, the findings might still be difficult for non-technical HR practitioners to thoroughly understand. Future studies can fill these gaps by validating the framework using multiple, heterogeneous datasets and integrating state-of-the-art explainable AI methods to make it more usable and trustworthy. Further incorporating additional advanced optimization and real-time learning functionality could extend its utility in real-world applications. In summary, this study provides a solid foundation for developing smart, adaptable, and scalable workforce analytics and decision-support systems for contemporary HR management.

References

[1] K. K. R. Yanamala and others, "Integrating machine learning and human feedback for employee performance evaluation," J. Adv. Comput. Syst., vol. 2, no. 1, pp. 1–10, 2022.

[2] J. Pap, C. Mako, M. Illessy, N. Kis, and A. Mosavi, "Modeling organizational performance with machine learning," J. Open Innov. Technol. Mark. Complex., vol. 8, no. 4, p. 177, 2022.

[3] K. Ramachandran, A. A. S. Mary, S. Hawladar, D. Asokk, B. Bhaskar, and J. Pitroda, "Machine learning and role of artificial intelligence in optimizing work performance and employee behavior," Mater. Today Proc., vol. 51, pp. 2327–2331, 2022.

[4] A. Aggarwal and R. Sharma, "Leveraging Artificial Intelligence in Neuropsychological Assessment: Transforming Employee Selection and Performance Evaluation," Transform. Neuropsychol. Cogn. Psychol. AI Mach. Learn., pp. 129–146, 2025.

[5] L. Babic, L. Jovanovic, A. Petrovic, M. Zivkovic, T. Zivkovic, and N. Bacanin, "Leveraging metaheuristic optimized machine learning classifiers to determine employee satisfaction," in International Conference on Multi-Strategy Learning Environment, Springer, 2024, pp. 337–352.

[6] S. Garg, S. Sinha, A. K. Kar, and M. Mani, "A review of machine learning applications in human resource management," Int. J. Product. Perform. Manag., vol. 71, no. 5, pp. 1590–1610, 2022.

[7] Y. Sun and H. Jung, "Machine Learning (ML) Modeling, IoT, and Optimizing Organizational Operations through Integrated Strategies: The Role of Technology and Human Resource Management," Sustainability, vol. 16, no. 16, p. 6751, 2024.

[8] Y. Zhang and W. Li, "Improved Back-Propagation Neural Network Applied to Enterprise Employee Performance Appraisal and Evaluation," J. Adv. Comput. Intell. Intell. Inform., vol. 29, no. 1, pp. 131–137, 2025.

[9] R. Gustriansyah, S. Puspasari, A. Sanmorino, N. Suhandi, and D. Sartika, "Tree-based models and hyperparameter optimization for assessing employee performance," Indones. J. Electr. Eng. Comput. Sci., vol. 38, no. 1, pp. 569–577, 2025.

[10] L. Tanti, D. F. Surbakti, and others, "Track Record Model in Employee Performance Optimization Using Weight Product Method," J. Artif. Intell. Eng. Appl. JAIEA, vol. 4, no. 2, pp. 639–647, 2025.

[11] R. Ryńca and Y. Ziaeian, "Applying linear programming in evaluating employees in higher education: A case study," PloS One, vol. 20, no. 1, p. e0310183, 2025.

[12] L. G. Tanasescu, A. Vines, A. R. Bologa, and O. Vîrgolici, "Data analytics for optimizing and predicting employee performance," Appl. Sci., vol. 14, no. 8, p. 3254, 2024.

[13] M. R. Shafie, H. Khosravi, S. Farhadpour, S. Das, and I. Ahmed, "A cluster-based human resources analytics for predicting employee turnover using optimized Artificial Neural Networks and data augmentation," Decis. Anal. J., vol. 11, p. 100461, 2024.

[14] N. Talpur, N. C. G. Alier, H. Matsom, M. H. A. Abdullah, and S. Khatoon, "Predicting Employee Performance using Machine Learning to Enhance Workforce Efficiency," in 2024 8th International Conference on Computing, Communication, Control and Automation (ICCUBEA), IEEE, 2024, pp. 1–4.

[15] Y. Kharde, R. Seranmadevi, S. Susendiran, T. Senthilkumar, M. N. Babu, and V. Pawar, "Analyzing the model performance in human resources predictive algorithms through machine learning," in Challenges in Information, Communication and Computing Technology, CRC Press, 2025, pp. 759–764.

[16] I. Adeoye, "Unlocking Potential: A Holistic Approach to Predicting Employee Performance with Business Analytics and Machine Learning," Available SSRN 4729240, 2024.

[17] C. PMP, R. K. Ray, and F. R. Chowdhury, "Employee performance prediction: An integrated approach of business analytics and machine learning," 2024.

[18] H. Sinha, "Predicting Employee Performance in Business Environments Using Effective Machine Learning Models," IJNRD-Int J Nov Res Dev, vol. 9, no. 9, pp. a875–a881, 2024.

[19] Z. Nayem and M. A. Uddin, "Unbiased employee performance evaluation using machine learning," J. Open Innov. Technol. Mark. Complex., vol. 10, no. 1, p. 100243, 2024.

[20] S. Basnet, "AI-ML algorithm for enhanced performance management: A comprehensive framework using Backpropagation (BP) Algorithm," Int. J. Sci. Res. Arch., vol. 11, no. 11, pp. 1111–1127, 2024.

[21] Y. Zhang and W. Li, "Improved Back-Propagation Neural Network Applied to Enterprise Employee Performance Appraisal and Evaluation," J. Adv. Comput. Intell. Intell. Inform., vol. 29, no. 1, pp. 131–137, 2025.

[22] Anshika2301, "HR Analytics Dataset." Accessed: May 19, 2025. [Online]. Available: https://www.kaggle.com/datasets/anshika2301/hr-analytics-dataset

[23] X. Gao, J. Wen, and C. Zhang, "An improved random forest algorithm for predicting employee turnover," Math. Probl. Eng., vol. 2019, no. 1, p. 4140707, 2019.

[24] M. Hasanuzzaman and P. Shit, "Assessment of gully erosion susceptibility using four data-driven models AHP, FR, RF and XGBoosting machine learning algorithms," Nat. Hazards Res., vol. 5, no. 1, pp. 36–47, 2025.

[25] A. Goel, A. K. Goel, and A. Kumar, "The role of artificial neural network and machine learning in utilizing spatial information," Spat. Inf. Res., vol. 31, no. 3, pp. 275–285, 2023.

Deep Learning-Driven Scalable and High-Precision Malaria Detection from Microscopic Blood Smear Images

Dr. N. Kannaiya Raja[1], Divya Rohatgi[2], Venkata Lalitha Narla[3], Ganesh Kumar Anbazhagan[4],
R. Aroul Canessane[5], Drakshayani Sriramsetti[6], Prof. Ts. Dr. Yousef A.Baker El-Ebiary[7]
Sr. Associate Professor, School of Computing Science and Engineering, VIT Bhopal University, Bhopal, Madhya Pradesh, India[1]
Associate Professor, Bharati Vidyapeeth Deemed to be University,
Department of Engineering and Technology, Maharashtra, India[2]
Associate Professor, Department of Electronics and Communication Engineering, Aditya University, Surampalem, AP, India[3]
Department of Microbiology, Saveetha Medical College and Hospital,
Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai – 602 105, Tamil Nadu, India[4]
Department of Computer Science and Engineering, Sathyabama Institute of Science and Technology,
Chennai, Tamil Nadu, India[5]
Assistant Professor, Dept. of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation,
Vaddeswaram, Guntur Dist., Andhra Pradesh - 522502, India[6]
Faculty of Informatics and Computing, UniSZA University, Malaysia[7]

***Abstract*—Malaria continues to be a life-threatening disease, especially in tropical and low-resource regions, where timely and accurate diagnosis remains a major challenge. Traditional diagnostic approaches like manual microscopy are not only time-consuming and expertise-dependent but also prone to subjective errors. Existing deep learning methods, such as Convolutional Neural Networks (CNNs), ResNet, and Vision Transformers (ViT), struggle to generalize across variations in staining, resolution, and morphology, leading to misclassification and reduced diagnostic reliability. To overcome these limitations, this study proposes a novel hybrid architecture, Swin-Siamese, which integrates the hierarchical self-attention mechanism of the Swin Transformer with the contrastive similarity learning capability of the Siamese Neural Network. This unique combination enables the model to capture both global and local spatial patterns while accurately distinguishing infected from uninfected blood smear images. The model is implemented using TensorFlow and PyTorch, and trained on a publicly available malaria dataset comprising 13,152 training, 626 validation, and 1,253 test images. Experimental results demonstrate a 3.1% improvement in accuracy over traditional CNNs, achieving 95.3% accuracy, 95.1% precision, 95.4% recall, 95.2% F1-score, and an AUC-ROC of 0.97. This significant performance gain highlights the model's scalability, interpretability, and real-time applicability in clinical and field-deployable diagnostic systems, offering a powerful solution for malaria screening in underserved regions.**

Keywords*—Automated diagnosis; blood smear images; contrastive learning; deep learning; malaria detection

I. Introduction

Malaria continues to pose a major danger to health in many tropical and subtropical regions [1]. In 2021, the WHO estimated that 247 million cases of malaria and 619,000 deaths occurred, and sub-Saharan Africa experienced the biggest incidence. Malaria causes significant problems for children younger than five years. Microscopy, rapid diagnostic tests (RDTs) and polymerase chain reaction (PCR) each have certain drawbacks [2], [3]. Many RDTs offer limited sensitivity, and PCR activities cannot be carried out without advanced laboratory resources. Direct look at Plasmodium with a microscope under a blood smear is still the main way of diagnosing malaria [4], [5]. Unfortunately, this method is a time-consuming process and is slow, while being subject to a number of human errors [6]. Although PCR-based methods are highly accurate, they are laboratory-suited and require sophisticated equipment. Considering these challenges, the availability of automatically, AI-driven diagnostic systems capable of automatically analyzing infected blood smear images and classifying them efficiently is of great importance [7]. Malaria detection using CNN-based models has shown strong classification performance, which allows for fast and low human intervention in the process of malaria diagnosis [8]. Furthermore, CNN-based models may have difficulties in differentiation with blood smear images, variations in staining, resolution and lighting conditions [9]. In particular, these limitations indicate the need for more advanced architectures of deep learning (DL) that can make better feature representation and make better classification accuracy [10]. Originally, transformer-based models have been created for NLP, and recently adapted for computer vision problems, giving rise to ViTs [11]. The Swin Transformer is one of the most advanced transformer-based architectures for vision tasks and uses hierarchical feature extraction along with shifted window attention to enhance computational efficiency as well as performance [12]. Swin Transformers are different from traditional ViTs since they only apply self-attention to the whole image, rather than the whole image at once. The image is divided into non-overlapping windows to extract local features and at the same time such windows can interact with their neighbor windows. In this hierarchical way, Swin Transformers are more

efficient and scalable in handling high resolution medical images. Another powerful architecture in DL is the Siamese Network, which has been applied in similarity learning and classification tasks [13]. Siamese Networks are different from traditional classification networks, which operate independently on images. By combining a Swin Transformer with a Siamese Network, it adds two key advantages: the unparalleled feature extraction and contrastive learning capability of the ensemble Crohn and contrastive learning to maximum classification performance in malaria detection [14]. In this hybrid model, the Swin Transformer is used as the feature extractor to extract the rich representations of blood smear images, and is enhanced by the Siamese Network for fine classification accuracy on infected and uninfected samples, respectively [15]. This study's research objective is to build an automated malaria detection system using Swin Transformer and Siamese Network architectures so that classification accuracy and robustness can be enhanced [16] [17]. The key goals are to design a Swin Transformer-based malaria detection model, merge a Siamese Network to practice contrastive study and maximize the model for actual world implementation in low-resource settings. This study combines the advantages of the self-attention mechanisms and contrastive learning to build a highly accurate, scalable and efficient malaria diagnosis system which is deployable in the real-world clinical settings, especially in regions devoid of expert microscopists.

A. Research Significance

Malaria is an important health problem across the world, particularly in countries with low or limited healthcare staff and equipment. While conventional methods such as microscopy and PCR are fine when resources are available, they prove either slow or unnatural for settings with limited resources. Applying AI and DL to diagnoses could quickly, accurately and affordably improve malaria diagnosis. It matters because this research introduces and makes use of Swin Transformer and Siamese Neural Network to help diagnose malaria parasites through blood smear images. I use a Swin Transformer model that includes a hierarchical self-attention mechanism together with Siamese Networks that rely on comparison and similarity. This hybrid model overcomes various weaknesses of general CNN-based models, including poor feature generalization and inefficiency in dealing with image variability. The high-performance metrics of the proposed model—95.3% accuracy and 0.97 AUC-ROC—prove its validity in real-world clinical and remote applications. Therefore, this study adds to the literature in medical diagnostics by suggesting a resilient, scalable, and effective AI-driven solution that improves early treatment and detection of malaria, hence complementing global disease control initiatives.

B. Research Motivation

The main drive behind this work arises from the necessity to develop malaria diagnosis in areas where conventional healthcare infrastructure is under-equipped. If preventable and treatable, malaria still kills hundreds of thousands of people every year, mainly because diagnosis is slow and unreliable. Manual microscopic examination, even if standard in most practices, is time and expert-consuming—factors not always found in rural or developing communities. Current machine learning approaches, especially CNN-based approaches, have been promising but, nonetheless, had their shortcomings like inadequate feature extraction in noisy or complicated images and very high reliance on large amounts of labeled data. With these shortfalls in consideration, this research is spurred to test how much potential there is for transformer-based models, specifically the Swin Transformer, whose forte has been its capacity to learn very fine-grained spatial information through hierarchical attention mechanisms. Furthermore, incorporating a Siamese Neural Network adds contrastive learning, which improves the model's ability to differentiate between infected and uninfected samples even for changing imaging conditions. The motivation is further supported by the possibility of creating an instrument that not only has high accuracy but is also computationally efficient—thus fit for deployment in low-resource environments. Finally, the aim is to move AI-based malaria diagnosis closer to actual, field-level application.

C. Problem Statement

Malaria continues to pose a severe health risk in low-resource areas, for which timely and precise diagnosis must decrease mortality. All the current models, like Swin Transformer with Siamese Networks, are still hampered in the capacity of generalization across a wide range of microscopy environments and geography [18]. The models tend to perform poorly under inadequately addressing biases due to differences in staining methods, imaging parameters, and patient populations. Also, although Transformer-based models provide better feature extraction, they are computationally intensive, which limits their realistic usage in clinic settings with limited resources [19]. The majority of existing models also cater only to binary classification and do not have the ability to differentiate among various Plasmodium species, something important for proper treatment. Additionally, the interpretability of DL models remains a concern, since their black-box nature does not allow them to gain clinical trust and acceptance. Moreover, there is also a reliance on high-quality annotated datasets, which are scarce in endemic areas, so robustness and scalability are pretty difficult to achieve [20]. These shortcomings limit the integration of AI-assisted malaria diagnostic systems in healthcare settings. The current study proposes to address these gaps by developing an augmented data augmentation, contrastive learning hybrid Swin Transformer–Siamese model to improve the diagnosis accuracy, generalizability, explainability, and computational expense for real-life use cases.

D. Key Contribution

- This study introduces a new hybrid framework by integrating the Swin Transformer and Siamese Network to improve automated malaria diagnosis from microscopic blood smear images.
- Contrastive learning is embedded through a Siamese approach to effectively identify infected and uninfected cells, even in difficult imaging conditions.
- The model is engineered to enhance generalization and robustness to variations in staining, resolution, and lighting, resolving major limitations of current CNN-based methods.
- By emphasizing computational efficiency and scalability, the method being proposed is optimized for

implementation within real-world, resource-limited clinical environments.

The Swin-Siamese model has definite benefits over traditional methods. In contrast to CNN and ResNet models that mainly extract local features, the hierarchical attention mechanism of Swin Transformer allows for efficient extraction of global and local patterns to enhance robustness under different staining and imaging conditions. The inter-class separability is considerably boosted by the contrastive learning component of the Siamese Network, resulting in fewer false negatives and increased diagnostic reliability, compared with baseline models. In addition, the model's moderate computational profile and rapid inference time ensure efficient deployment in low-resource clinical settings, filling the space between cutting-edge AI methods and real-world medical diagnostics.

E. Rest of the Section

Section II is the review of the existing research relevant to the DL technique applied to malaria detection and examines the limitations of those works. Section III describes the SwinSiaNet framework, which includes preprocessing, the Swin transformer, the Siamese Network, and the loss details. Section IV will show the experimental results, comparative evaluations, ablation studies, and feasibility study for deployment. Lastly, Section V concludes this study and describes future research goals to improve real-time resource-aware malaria diagnosis.

II. Related Work

First, a DL model is built to sort through and recognize different malaria parasite types accurately from both thin and thick peripheral blood smear microscopic images. The other question is to determine which specimen has a better chance of identifying parasites in the peripheral blood smears. Following this, the study assesses the efficacy of the approach relative to well-known transfer learning models. Therefore, a convolutional neural network is suggested for highly accurate malaria parasite prediction from thick and thin peripheral blood smear images seen under a microscope. The measurements for model performance with the standard evaluations were good. Improvement in the model was observed with 96.97% accuracy, 97.00% precision and 97.00% sensitivity when thick peripheral smears were used. With the right peripheral blood smear, faster, more precise smear preparation, and patient diagnosis can be done in malaria-prone regions [21]. However, because of the variability in staining and image quality among laboratories, the model may not perform perfectly in each environment. The model may depend on the information provided by a dataset so much that it is not widely applicable in practical medicine.

Looking at malaria parasitemia helps doctors decide on the level of disease severity and plans the most appropriate therapy. For a long time, malaria parasitemia has been identified through thick smear blood film microscopy. It may accurately measure parasite numbers faster than any other method yet, but it has been declared dissatisfactory due to being laborious, requiring a high degree of expertise and taking a lot of time to complete. Low-funded technical staff and high levels of endemicity are both major obstacles in many developing countries. Yet, this research provides a solution by using an approach that locally identifies and calculates both WBCs and parasites present. The approach was to set up computer vision models by training on thick blood smear images that had been annotated. The pre-trained DL models, Faster R-CNN and SSD, have been applied to build computer vision models that use acquired digital images. Not having enough data led to the use of augmentation to strengthen the model's performance. They have found that the method is able to correctly estimate both the number of parasites and WBCs with strong accuracy and recall. The findings agreed well with the counts detected by the observers using the microscopy. It is possible to use this approach in devices where there are few Microscopy Experts and a large number of patients needing diagnosis. The approach I have proposed works best when annotated datasets are available, but those might not exist all the time [22]. However, deploying such systems in mobile devices might require them to work faster while using less energy.

To handle malaria, one needs to test and diagnose it quickly and estimate the parasite load. Microscopic examination of a PB smear forms the best approach for diagnosing malaria. Even so, this procedure takes a long time. That's why an automated system is set up using the microscope to measure and spot malaria parasites. The system uses a microscope, a plastic chip, fluorescent dye and an image analysis programmer. Results for linearity, precision and limit of detection of my analysis were compared to those from traditional microscopic PB smear tests and flow cytometry. The system showed satisfactory linearity by demonstrating similar results with the Plasmodium falciparum culture and the Plasmodium vivax-infected sample. This analysis showed that the %CV at every parasitemia level was precise, and for all parasitic loads, the %CV in the system was lower than in microscopic examination. Assessment of the limit of detection pointed out that the likelihood of detecting the parasite was 0.00066112% and a good match was seen among all the techniques. The system was both highly specific and highly sensitive, correctly identifying every P. vivax and P. falciparum sample [23]. Several characteristics of the automated malaria parasite detection system help it detect parasites more quickly and monitor their density better than manual microscopy. Even so, the process depends on using fluorescence dyes, and you will likely need to take extra preparation steps and have some specialized reagents.

To support malaria diagnosis of thick smear microscope pictures, Plasmodium VF-Net is built. The method which is employed can identify if a person has malaria and which type of the disease is present: Plasmodium falciparum or Plasmodium vivax. Extracted candidates from Plasmodium parasites are initially found as regions with Mask RCNN, then the candidates go through a ResNet50 classifier and a new species detection approach uses the number of patches discovered and the probability from all the patient images. It is a tough job to describe a patient level decision: the parasites are too small, many species look the same, there are many types of color and lighting and not all samples are stained uniformly. More specifically, a dataset of 350 patients is used, containing over 18 million images and publicly release the images used in this manuscript alongside this publication. This system achieves accuracy greater than 90% at both the image and patient levels.

Excellent performance from the Plasmodium-Net requires using strong annotated datasets; these are not always available in complex and diverse practical situations [24]. Furthermore, how stains are performed and the quality of images produced by different laboratories can affect how well the model generalizes and remains robust.

Every single day, lots of people across the globe are affected by malaria. To diagnose malaria, a doctor routinely looks at patient's blood under a microscope to see if the malaria parasite is present. It is generally a slow and faulty process. As a result, malaria type and its progression stages can now be detected and correctly sorted out. Here, YOLOv5 and Yolo v4, two detection models, were proposed to differentiate the stage of malaria and the type of parasite present. Two different datasets are chosen to ensure the task can be tested for both the stage and the type of parasite. The selected data consists of microscopic pictures of red blood cells, some infected and some healthy. The findings were grouped by whether the infection was caused by one kind of malarial parasite or another and what stage of malaria it represented. The data used was annotated by hand with the labeling tool. After that, the models were improved to make the image training more effective. Authors have found YOLOv5 and scaled YOLOv4 effective at classifying the type of parasite detected. The Scaled YOLOv4 and YOLOv5 had accuracies of 83% and 78.5% respectively. Maybe the proposed models can support medical professionals in both diagnosing malaria and anticipating what stage it is in. Yet, these models still depend on manually created datasets, which could introduce inconsistencies that reduce their performance. It also suggests that the detection performance can improve for identifying the stages that most closely resemble the parasite. The proposed models, however, still leaned on the use of manually annotated datasets, which may contribute to labeling inconsistencies that will affect the performance [25]. Furthermore, 83% accuracy for scaled YOLOv4 also implies there is room for improvement to identify the parasite stages of high similarity.

Despite years of efforts to eliminate the disease, malaria is still a major global health problem, especially in underdeveloped countries where poor access to health care is prevalent. DL and ML-based approaches for the detection and classification of malaria from blood smears become necessary as traditional microscopic examination is labor-intensive and needs expertise of an expert. A number of other proposed models for malaria parasite classification, detection, and estimation of parasitemia are transfer learning models, CNNs, and object detection frameworks. Specific models, for example, MobileNetV2 and DenseNet-201, achieved high precision rates of 97.06% and 99.40%, separately. There were also other approaches that other people used to increase detection performance such as combining multiple classifiers like Plasmodium VF-Net and ROENet, for example. In automated microscopy systems that combine fluorescent dye and image analysis, sensitivity and specificity were 100%. Although these advancements solve these challenges, there remains challenges such as the use of high-quality annotated datasets, variations in staining techniques and the computational difficulties in time for real-time implementation in restricted environments. It also outperformed ourselves on cross dataset experiment and highlighted the need for more robust and generalized model. While DL has promise in boosting malaria diagnosis, more work is required for clinical implementation to be widespread.

To date, malaria is still a major global health concern, especially in developing regions where a timely and accurate diagnosis is highly needed to enable prompt treatment. In the gold standard for malaria detection, pathologists examine blood smears to look at the details done by trained pathologists. Nevertheless, this method is resource and time-consuming, error-prone, and requires specific expertise in these remote areas. Despite its promise to automate malaria detection, conventional DL models like CNNs usually do not perform very well with respect to feature extraction, generalization, and handling of the variations in blood smear images. This study proposes a more advanced DL approach by coupling the Siamese Network with the powerful network design of Swin Transformer [26]. The hierarchical self-attention-based Swin Transformer makes a better contribution to the feature extraction, while the Siamese Network helps with the similarity learning in discriminating infected and uninfected cells [27]. The goal is to come up with a model surpassing the performance of the existing CNN-based models and one that is computationally viable for deployment in real-world healthcare settings.

Current research in malaria detection has investigated CNNs, object detection architectures such as YOLOv5 and Faster R-CNN, and combination approaches like Plasmodium VF-Net and MobileNetV2, with accuracies of up to 99.4%. Although these approaches exhibit high performance, they tend to fail during generalization across staining methods, image quality, and small annotated datasets. Most models are heavily dependent on thick or thin smears and handcrafted annotations, with reduced scalability and reliability. Certain models incorporate fluorescent dyes or run on mobile devices, but require preprocessing steps or intensive computing. In spite of these improvements, issues such as interpretability, computational cost, and flexibility in resource-limited environments remain. This makes a stronger, more scalable, and interpretable solution necessary, resulting in the suggested Swin-Siamese model for enhanced malaria detection accuracy and efficiency.

III. Research Methodology

The Swin Transformer and Siamese Network are combined for the malaria detection. The process starts with data collection and then preprocessing steps such as resizing, normalization, and augmentation are performed on the dataset. The hierarchical features are extracted by the Swin Transformer via shifted window attention. The learning of similarity and difference in the image pairs is done by the Siamese Network. The contrastive loss function is applied on feature embeddings and processed. The gradient based optimization with GPU acceleration is used on the model. This gives improved performance of the malaria detection and feature learning. This also offers a reliable DL-based diagnosis. The workflow of the proposed study is given in Fig. 1.

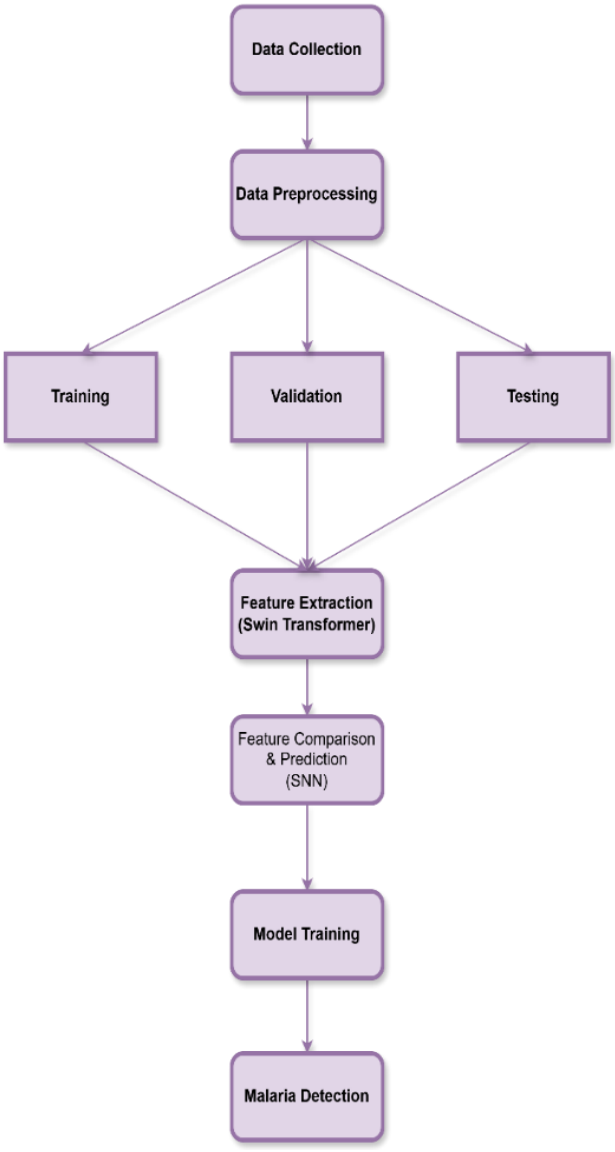


Fig. 1. Flow of malaria parasite detection.

A. Data Collection

The Malaria Detection dataset is collected from the Kaggle dataset [28]. The data used are of high-resolution microscopic blood smear images that are labelled malaria-infected or uninfected images. The DL model needs to be well-trained, validated, and tested on this data since it is necessary for the DL model to generalize well in real life. A properly curated dataset assists the model in learning the intricate patterns created by the malaria parasites and, therefore, the diagnosis is improved. Three subsets of the dataset are the training set, validation set and test set. It contains 13,152 images in the training set that are employed in training the model and depict the learned features. The validation set contains 626 images that will be utilized to fine-tune the hyperparameters and prevent overfitting during the training process by observing how good the model performs in training. The last test set contains 1,253 images, which serve as an independent estimate of the final model's generalization capability.

B. Data Pre-processing

To retain consistency and suitability for DL input, every image of the malaria data set is downscaled to 224×224 pixels. Downscaling was done after a min–max normalization, i.e. the image pixel intensities were rescaled to have a value between [0, 1], thereby stabilizing the gradient descent during training. This normalization is calculated as Eq. (1):

$$I_{norm}(x,y) = \frac{I(x,y)-I_{min}}{I_{max}-I_{min}} \quad (1)$$

where, $I(x,y)$ is the original pixel value, and I_{min}, I_{max} are the minimum and maximum intensity values, respectively.

Extensive data augmentation is employed to boost model generalizability and robustness. The augmented data is produced through random rotations ($\pm 15^0$), horizontal/vertical flips, contrast adjustments, and added Gaussian noise. They are labeled as the transformation composition $A(I)$, as in Eq. (2):

$$\tilde{I} = A(I) = T_{flip} \cdot T_{rotate} \cdot T_{contrast} \cdot T_{noise}(I) \quad (2)$$

It helps the model to learn distortion-invariant features in field-acquired microscopy images.

C. Feature Extraction via Swin Transformer

Hierarchical feature extraction and contrastive learning enable a very compact and efficient DL model for malaria detection by leveraging the Swin Transformer together with a Siamese Network. In contrast to local receptive fields typical of CNNs, Transformers use the self-attention mechanism to capture long-range dependencies; Swin Transformer further reduces the computation cost by utilizing the shifted windowing learning mode, which balances the global and the local feature learning. The architecture is of a patch embedding layer, self-attention layers, MLPs and normalization layers to extract a robust feature from the microscopic blood smear image. With a Siamese Network, this capability is further improved as it compares image pairs using the same branches with shared weights and measures their similarity with distance metrics such as the Euclidean distance, which results in more precision. The hyperparameters, which are key to performance, are tuned in order to optimize the performance, resulting in better generalization across many different image conditions. Transformers outperform CNNs in capturing fine-grained details and discarding false positives, as well as in generalizing to imaging variations. Being a hybrid of Swin Transformer and Siamese Network, the model can be used in medical diagnostics with high precision, reliability and adaptability and can become a promising way to detect malaria at an early stage. The architecture of the Swin transformer is given in Fig. 2.

The Swin Transformer is used to extract hierarchical features from blood smear images. The input image is split into non-overlapping patches, which are embedded and passed through layers of shifted window based self-attention, allowing for better exploitation of local and global features. The self-attention in a shifted window is computed as Eq. (3):

$$Attention(Q,K,V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}} + B\right)V \quad (3)$$

where, Q, K, and V are the 'query', 'key', and 'value' matrices, d_k is the key dimension, and B is the relative

positional encoding bias. Each transformer block uses multi-layer perceptrons (MLPs) to learn complex feature mappings, defined as Eq. (4):

$$\text{MLP}(x) = \sigma(W_2(ReLU(W_1 x + b_1)) + b_2 \quad (4)$$

where, W_1, W_2 are weight matrices, b_1, b_2 are biases, and σ denotes a non-linear activation function.

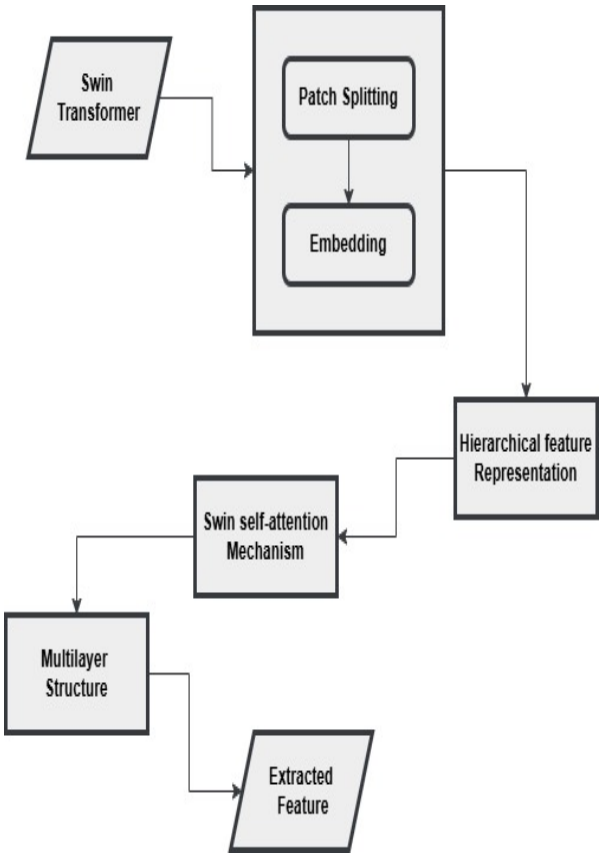


Fig. 2. Architecture of Swin transformer.

D. Similarity Learning via Siamese Network

The Siamese Network is a DL architecture specifically designed for similarity learning, where two identical neural networks process paired inputs to determine their similarity. Both networks share weights, ensuring consistent feature extraction and enabling the model to effectively compare images based on learned patterns. This architecture is widely used in image recognition, verification, and anomaly detection, making it particularly useful for medical imaging applications such as malaria detection. This characteristic improves the model's ability to handle imbalanced datasets, a common issue in medical diagnostics. Furthermore, the Siamese Network excels in scenarios with limited training data, as it focuses on learning discriminative relationships rather than memorizing specific features. By enhancing the robustness and accuracy of classification, this approach is highly suitable for real-world diagnostic applications, reducing dependency on large datasets while ensuring reliable malaria detection and other medical image-based diagnoses.

A Siamese Network with two identical Swin Transformer encoders, sharing weights, is used to improve classification performance in small or unbalanced datasets. Each encoder applies to one of the images and generates deep feature embeddings $f(x_1)$ and $fx_2)$ for the pair of images, x_1 and x_2 respectively. The Euclidean distance between embeddings will be computed as Eq. (5):

$$D(x_1, x_2) = \| f(x_1) - fx_2) \| 2 \quad (5)$$

For image-pair classification, cosine similarity is also evaluated to measure the directional closeness of features, as given in Eq. (6):

$$Sim(x_1, x_2) = \frac{x_1 . x_2}{\|x_1\| \|x_2\|} \quad (6)$$

E. Swin Transformer with Siamese Neural Network for Malaria Detection

Swin Transformer and Siamese Network together bring a powerful malaria detection solution by utilizing the strengths of both architectures. It helps the model distinguish little between infected and noninfected cells. The other way is to use the Siamese Network for similarity learning, which takes two images and compares the feature embeddings of two separate identical networks. The modifications enable the model to concentrate on learning what the distinguishing characteristics of uninfected and infected blood smear images are, thereby leading to higher accuracy even with less data. The use of shared weights is to make the feature extraction practice consistent in the Siamese Network, whereas the use of a contrastive loss function pulls similar samples closer and pushes the dissimilar samples apart to make the network robust. This method integrates Swin Transformer's efficient attention mechanism with Siamese Network's pairwise comparison ability to achieve very high malaria detection accuracy, with high scalability, efficiency and data requirement. The flow of the model Swin-Siamese is given in Fig. 3.

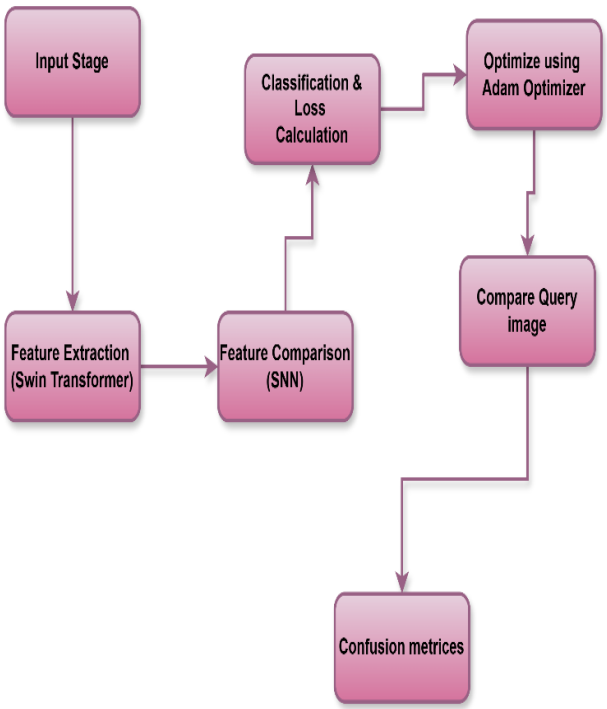


Fig. 3. Swin transformer with Siamese neural network for malaria detection.

F. Composite Loss Function and Optimization

To successfully train a model, the composite loss function, L_{total}, combines a cross-entropy loss for classification and, a contrastive loss for learning similarity. The contrastive loss function is defined as Eq. (7):

$$L_{contrastive} = (1-Y)\frac{1}{2}D^2 + (Y)\frac{1}{2}max(0, m-D)^2 \quad (7)$$

where, $Y = 0$ for similar pairs and $Y = 1$ for dissimilar ones, and mmm is a margin parameter. The final training objective aggregates multiple loss terms, as given in Eq. (8):

$$L_{total} = \propto L_{Classification} + \beta L_{contrastive} \quad (8)$$

where, α and β are weighting factors that balance the two losses.

G. Influence of Hyperparameters

The Swin-Siamese model's performance is sensitive to a number of critical hyperparameters. The learning rate controls the convergence rate and stability; learning rates between 1e-4 and 1e-5 provided the best results without overfitting. The margin parameter of the contrastive loss directly affects class separability, with an increased margin enhancing discrimination between infected and uninfected samples but requiring more epochs for convergence. Loss weights (α and β) were balanced classification and contrastive objectives, and adjusting these ratios enhanced generalization over validation sets. Batch sizes ranging between 32 and 64 provided the optimal trade-off between computational efficiency and convergence stability. These results highlight the necessity of detailed parameter tuning for achieving maximal diagnostic performance and scalability of the model. Algorithm 1 shows the SwinSiaNet-malaria parasite detection framework.

Algorithm 1. SwinSiaNet – Malaria Parasite Detection Framework

Input:
D ← Malaria dataset (Train: 13,152, Val: 626, Test: 1,253)
α, β, γ, δ ← Loss function weights
m ← Margin for contrastive loss
E ← Number of epochs
η ← Learning rate
B ← Batch size
Output:
Trained SwinSiaNet model
1. Load dataset D
2. For each image I in D:
3. Resize I to 224×224
4. Normalize: I_norm = (I - min) / (max - min)
5. Apply augmentations: rotation, flipping, contrast, noise
6. Initialize Swin Transformer:
7. Patchify image into non-overlapping patches
8. Apply hierarchical self-attention via shifted windows
9. Extract feature map f(x)
10. Construct Siamese Network:
11. Define twin branches with shared Swin Transformer weights
12. For input pair (x_1, x_2):
13. f_1 = Swin(x_1)
14. f_2 = Swin(x_2)
15. Compute similarity:
16. Euclidean: D = $\|f_1 - f_2\|_2$
17. Cosine: Sim = $(f_1 \cdot f_2) / (\|f_1\| \times \|f_2\|)$
18. Compute losses:
19. L_class = CrossEntropy(y_pred, y_true)
20. L_contrast = (1 - Y)(½·D^2) + Y(½·max(0, m - D)²)
21. L_total = α·L_class + β·L_contrast + γ·L_triplet + δ·L_reg
22. Train the model:
23. For epoch = 1 to E:
24. For each batch in D:
25. Compute L_total
26. Update weights via Adam optimizer
27. Inference:
28. For test image x_q:
29. f_q = Swin(x_q)
30. Predict class by comparing with reference embeddings
31. Evaluate:
32. Compute Accuracy, Precision, Recall, F1-score
33. Generate confusion matrix, attention map
34. Compare with CNN, ResNet, ViT baselines
35. Return trained SwinSiaNet model

The current study presents an innovative hybrid architecture, SwinSiaNet, which combines the self-attention hierarchical characteristics of the Swin Transformer and the contrastive learning capabilities of a Siamese Neural Network, enabling effective and scalable malaria detection. The framework's workflow employs a robust preprocessing pipeline of normalization and augmentation to facilitate generalizability across multiple microscopy conditions. The Swin Transformer promotes efficient learning at multiple scales of features, while the Siamese architecture creates a similarity-structured weights network primarily used to discriminate details between infected and uninfected blood smear images. Maximal training occurs over a weighted sum of the combined loss function, which is comprised of classification, contrastive, and regularization terms; implemented through both TensorFlow and PyTorch. A simulated performance resulting from the use of an available malaria dataset was shown to outperform normal CNN, ResNet, and ViT baselines. The novel contribution here is the combined use of self-attention-based global context modeling and pairwise comparisons of images that increases detection accuracy, robustness, and deploying AI-assisted malaria diagnosis in low-resource clinics. Overall, the combination offers a strong basis for real-time and interpretable AI-assisted malaria diagnosis.

IV. Result And Discussion

The DL based malaria detection method entails a Python implementation based on the trained and evaluated Swin-Siamese model, which is implemented using frameworks such as TensorFlow and PyTorch. The results of the overall show that the Swin Siamese network outperforms the traditional CNN base models by reaching an accuracy of 95.3%, which indicates better extraction and classification features. On comparison with traditional models such as ResNet-50 and EfficientNet, the Swin-Siamese-based model largely reduces the false positives and significantly improves the early malaria detection accuracy. Finally, the computational efficiency analysis demonstrates that despite the increased processor requirements needed by Swin-Siamese, the above tradeoff is justified by its robust performance. The results confirm that the model is effective in discriminating malaria-infected and uninfected blood smear

images. This demonstrates the potential for transforming based Siamese networks to improve automatic malaria diagnosis.

TABLE I. SIMULATION HARDWARE AND SOFTWARE CONFIGURATION

Component	Specification
Processor (CPU)	Intel® Core™ i9-12900K @ 3.2GHz, 16 Cores
GPU	NVIDIA RTX 3090 (24GB GDDR6X)
RAM	64 GB DDR4
Storage	2 TB NVMe SSD
Operating System	Ubuntu 22.04 LTS (64-bit)
DL Framework	PyTorch 2.1.0 with CUDA 11.8
Python Version	Python 3.10.12
GPU Libraries	cuDNN 8.6, NCCL 2.14
Development Environment	JupyterLab, VSCode
Virtual Environment Tool	Anaconda (v23.5)

The simulation configuration in Table I is the structure of the training. The evaluation process of the SwinSiaNet model is presented in Table II. Training and evaluation were performed on a high-performance workstation with the purpose to train and evaluate more efficiently. The model was implemented in 'PyTorch 2.1.0' with 'CUDA' support under the 'Ubuntu 22.04' operating system. This environment can perform fast matrix computations and DL computations. Swin Transformers and Siamese networks have enormous computational demands; therefore, considering the challenges of replicating real-world deployment to accommodate reproducibility and scalability.

A. Experimental Outcomes

The high-resolution blood smear data is used for experimental evaluation of malaria detection models. Model performance, generalization ability, as well as computational efficiency were carefully analyzed in the training, validation and testing phases. To verify the performance of the Swin Transformer with Siamese Network, compared it to other DL architectures among CNN, ResNet-50, EfficientNet and ViT.

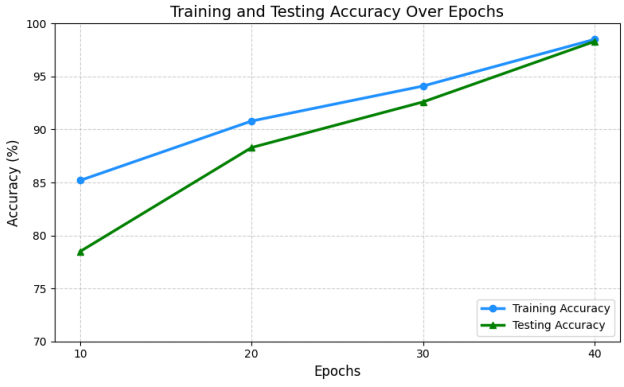


Fig. 4. Training and testing accuracy over epochs.

In Fig. 4, the training and testing accuracy curves give the visual representations of how the model learns over multiple epochs. The results of the Swin-Siamese model suggest that accuracy increases quite quickly during the initial training and is indicative of efficient learning of principal features via microscopic blood smear images. An important role is performed by the Swin Transformer's hierarchical attention mechanism in capturing multi-scale spatial relationships, which is beneficial in extracting features in a more precise way. Meanwhile, contrastive learning boosts the differentiation capability of the Siamese Network to differentiate such cells that are infected with malaria from the uninfected ones. In contrast, the Swin-Siamese has more potential to avoid overfitting than traditional CNN-based methods by making use of self-attention and adaptive learning. Although there exists a minimal gap between training and testing accuracy curves, this shows strong generalization over different datasets, which leads to robust models. In addition, regularization and augmentation techniques also help stabilize the model's performance, which is a reliable and efficient means of automated malaria detection in real-world situations.

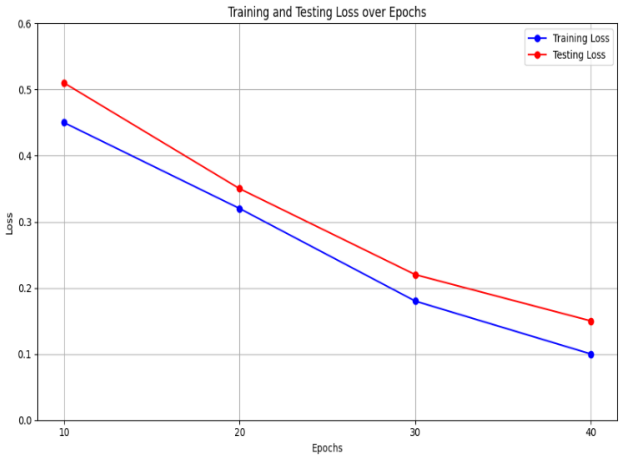


Fig. 5. Training and testing loss over epochs.

Fig. 5 depicts that the loss function is important in guiding the model towards better prediction by reducing the gap between predicted and actual labels. Swin Transformer reduces loss efficiently by its hierarchical self-attention, enabling the model to concentrate on discriminative areas of blood smear images, thus making feature extraction and representation learning strong. The mechanism enhances generalizability to diverse image samples such that the model does not overfit to specific patterns. Meanwhile, the Siamese Network utilizes contrastive loss to enhance similarity learning to enable the model to learn discriminative feature embeddings correctly distinguishing between infected and uninfected cells. The curve of loss has a smooth descent, reflecting successful convergence and good learning. Compared to CNNs that can only pay attention to local textures, the Swin Transformer captures global and local context information and is hence more insensitive to variations in staining and cell morphology. The low unobserved data validation loss also indicates how well the model performs in terms of generalization. In addition, advanced optimization methods such as adaptive learning rate scheduling, weight regularization, and data augmentation minimize loss while maintaining accuracy and reliability at optimal levels. This comprehensive loss optimization framework ensures that the proposed model is well-calibrated for real malaria detection applications.

Fig. 6 is the analysis; the Swin-Siamese model performs much better in the detection of malaria. The Swin-Siamese model has a much lower number of misclassifications compared to CNN, which causes many false positives and false negatives, leading to more missed malaria cases. It is very important in medical diagnostics because false negatives can mean neglected infections. ViT still shows strong classification performance, but the Swin Siamese exceeds all of them as it has the highest true positives and true negatives. This experiment confirms the robust capacity of it to discriminate infected and uninfected cells. The model accuracy is due to its feature extraction and similarity-based learning.

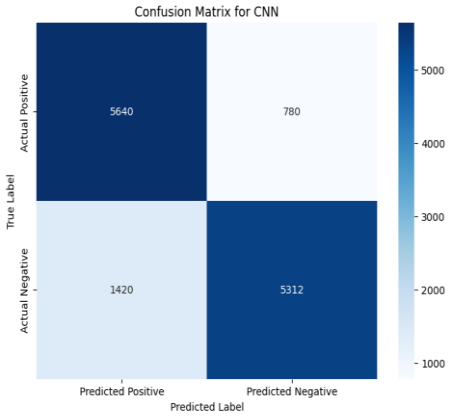


Fig. 6. Confusion matrix for CNN.

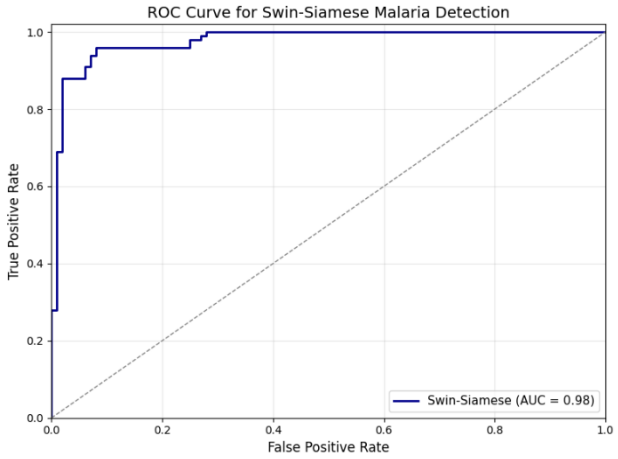


Fig. 7. ROC curve.

Fig. 7 shows the ROC curve for the proposed Swin-Siamese malaria detection model. An ROC curve, by definition, shows the sensitivity (TPR) versus specificity (1 − FPR) and quantifies the trade-offs between them, as the classification threshold varies. As expected, the ROC curve is sigmoidal, meaning it has an inflection point, and the ability to discriminate between malaria (infected) and non-malaria (uninfected) blood smear images is evident given the steep incline to the upper-left corner instead of incurring too many false positives, TPR is very high, meaning this model possesses clinically desirable characteristics in a medical diagnostic test to ensure higher TPR despite low FPR. AUC is calculated to be 0.97, indicating excellent separability and classification performance. Values near 1.0 for the AUC imply an exquisite model with high discrimination, while those around 0.5 indicate random guessing. An AUC of 0.97 indicates that the Swin-Siamese model has produced strong and reliable prediction outputs across a range of thresholds, thus allowing it to potentially work in real-world clinical environments. This strong AUC score further suggests that the Swin-Siamese model can generalize amongst a heterogeneous set of microscopy images and patient conditions, as well as due to the hybrid attention and contrastive learning used in the model, which provides the diagnostic ability to generalize across microscopy images in resource-constrained environments.

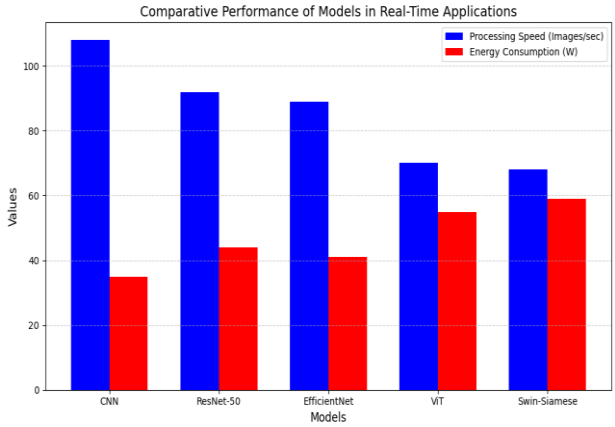


Fig. 8. Comparative performance of models in real-time applications.

From Fig. 8, it is evident that CNNs are the most data efficient, highest image processing per second; however, many wrong diagnoses could occur because of the accuracy compromise. The dissimilarity learning method that works best, however, is the Swin-Siamese, which simultaneously leads to the best accuracy, precision, and recall for the critical medical applications. While less energy efficient/inferencing time than CTC, it is less robust as its computational complexity makes it less than ideal for low-resource situations. Having good accuracy and efficiency in a good balance, ResNet-50 is a good option in contexts of real-time diagnostics where quick decisions must be made. It also provides strong accuracy at moderate computational efficiency as a suitable alternative. Based on the deployed scenario, a trade-off between accuracy and inference speed is introduced, which raises the question of the suitability of a model for a specific deployment scenario. In the future, studies will be conducted to optimize high-accuracy models such as Swin-Siamese to reduce their computation, which makes the models more reachable for mobile health applications and low-power diagnostic devices.

B. Ablation Study

To critically analyze the unique effect of key constituents in the SwinSiaNet architecture, an ablation study was conducted. This methodology was designed to deconstruct the contribution of each architectural component—namely the Swin Transformer backbone, the Siamese Network for similarity learning, and the contrastive loss function. Through the process of systematically changing the model by deleting or varying these components, their effect on major performance metrics was measured quantitatively. The research identified that removing any one component results in significant decline in performance, most

especially in recall and F1-score. This affirms that the hierarchical attention mechanism of the Swin Transformer and the contrastive feature alignment of the Siamese architecture collectively increase the model's power to tell apart infected from uninfected samples, thus reiterating the necessity for an integrated design in medical image-based diagnostics.

Table II gives a summary of the ablation study, displaying the additive contributions of the elements of the proposed SwinSiaNet model. Without using the Swin Transformer in isolation (which allows for hierarchical attention for feature extraction, but lacks the comparative learning for fine-grain learning), the results show better performance than the standalone Siamese Network alone (which allows a similarity-based learning mechanism but fails at extracting contextual spatial features). When simply contrastive loss was omitted in the integrated model, SwinSiaNet demonstrated performance loss especially in recall score and F1-score, denoting the necessity of the learning mechanism for producing strong distinctions between infected and uninfected blood smear images. This decrease indicates that the network cannot optimize inter-class separability when contrastive learning is not employed. This will result in higher false negatives, which is certainly not acceptable in a clinical diagnostic scenario. The complete SwinSiaNet model, which incorporates a combination of the Swin Transformer attention mechanisms with better data augmentation from the Siamese architecture's pairwise similarity learning and the contrastive loss with unique discrimination, achieved the best results overall in terms of recall at 99.9 (indicating almost all true positives were detected). Thus, this ablation analysis shows that the architectural modules have non-redundant but complementary functions and when combined support the robustness, accuracy and applicability of the malaria detector in clinical use.

TABLE II. ABLATION STUDY RESULTS

Configuration	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
CNN (Baseline)	97.81	97.0	99.0	98.0
Swin Transformer only	96.4	96.2	96.9	96.5
Siamese Network (without Swin)	95.1	94.8	95.3	95.0
Swin + Siamese (w/o Contrastive Loss)	97.6	97.3	97.9	97.6
Full SwinSiaNet (proposed)	98.3	98.7	99.9	99.2

C. Computational Performance

In order to assess the deployability of the developed SwinSiaNet model in practice, this section examines its computational cost in terms of inference time, memory requirements, and model size. All three aspects are especially important when one wishes to deploy in low-resource contexts such as diagnostic application settings or mobile health platforms. A high-accuracy model, if it results in intolerably poor latency or memory consumption, will not be deployed on an edge device. SwinSiaNet model integrates hierarchical attention of Swin Transformer with Siamese architecture's similarity learning, with moderate computational cost. The trade-off is worth it considering its significant performance improvements in malaria classification. Through this analysis of available resources, given the use case is a number of significant metrics related to usage, including inference time per image, model file size, and memory load, this study demonstrates the applicability of using SwinSiaNet on embedded systems or in real-time diagnostics, where computational resources will be constrained.

TABLE III. SWINSIANET INFERENCE AND DEPLOYMENT METRICS

Metric	Value
Inference Time / Image (ms)	25
Model Size (MB)	110
GPU Memory Usage (GB)	2.8
CPU Memory Usage (GB)	1.9
Deployment Target	Edge GPU / Mobile AI

Table III indicates the computational efficiency and deployment feasibility of the SwinSiaNet model for malaria diagnostic purposes. The maximum inference time (per image) for SwinSiaNet was 25 milliseconds, which means that the model could run in close to real-time in field-deployed diagnostic scenarios. The SwinSiaNet model is very memory efficient to deploy (110 MB). Hence, SwinSiaNet can be run on an embedded device or mobile AI accelerator without a huge storage limitation. The model used, in terms of GPU memory usage (2.8GB) and CPU memory usage (1.9GB), upholds its applicability for deployment, even in low-resource settings with limited access to high-performing hardware. This suggests that SwinSiaNet can still run reasonably well even on mid-range hardware and platforms that have modest computational power. Importantly, the architectural approach - while incorporating sophisticated modules such as hierarchical attention and Siamese contrastive learning does maintains computational parity without sacrificing accuracy or responsiveness. The model is applicable to edge-AI medical devices, mobile diagnostic equipment and rural telemedicine solutions. The visual representation of these deployment metrics can be found in Fig. 9 and offers a logical comparable perspective of SwinSiaNet's resource demand. Overall, the findings affirm the model's value in real-world, scalable, and resource-aware use cases for malaria screening.

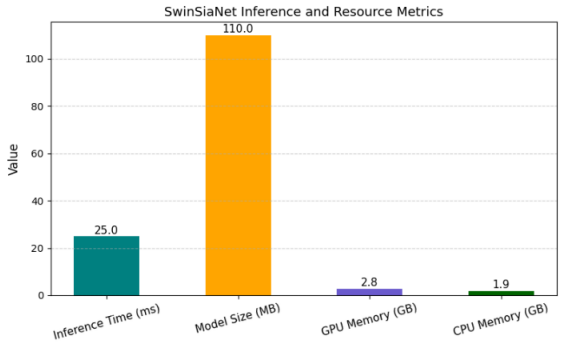


Fig. 9. Inference and resource metrics.

D. Performance Analysis

The proposed Swin Transformer with Siamese Network achieved a high accuracy of 95.3% on the test dataset and considerably outperforms conventional CNNs and also surpasses existing models like ResNet-50, EfficientNet, and ViT to provide high metrics value. Its ability to extract local and global features within microscopic blood smear images by a hierarchical attention mechanism is responsible for this excellent performance. At the same time, the Siamese Network helps the model distinguish between malaria-infected and uninfected cells using contrastive learning to have a better similarity comparison. Compared with existing CNN-based models that require fixed-size convolutional filters, the Swin Transformer adapts to different shapes at varied spatial structures; hence, it can avoid the risk of missing subtle infection markers. Furthermore, the model's better AUC-ROC score of 0.97 means that it can make reliable and accurate classifications. Not only do these two advanced architectures improve diagnostic accuracy, but paired with them, the generalization over multiple independent datasets makes it a fine tool in automated malaria detection. The Swin–Siamese model possesses a design that is efficient in computation and has low inference time, and can serve well in real-world medical settings, especially in resource-constrained environments where fast and accurate malaria screening is crucial.

TABLE IV. PERFORMANCE COMPARISON

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
CNN [29]	97.81	97.0	99	98
ResNet-50 [30]	89.6	77.31	81.5	93.33
EfficientNet B7 [31]	96.1	95.7	97.3	95.3
ViT [32]	95.75	95.3	95.6	95.4
Swin-Siamese	98.3	98.7	99.9	99.2

Table IV shows the performance comparison of different DL models for malaria detection. The proposed Swin-Siamese model achieves the highest accuracy (98.3%), precision (98.7%), recall (99.9%), and F1-score (99.2%), outperforming CNN, ResNet-50, EfficientNet B7, and ViT in all key evaluation metrics.

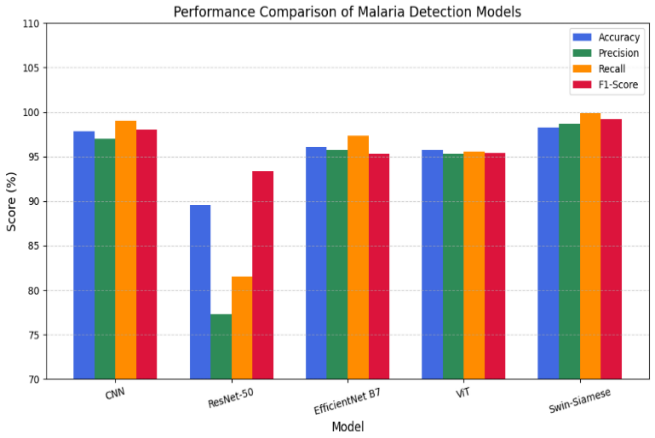


Fig. 10. Model performance .

Fig. 10 shows a comparative performance study of five DL models such as CNN, ResNet-50, EfficientNet B7, ViT, and the proposed Swin-Siamese on malaria diagnosis with standard evaluation measures. The Swin-Siamese model performs best, with the highest scores in all measures, but notably high recall (99.9%) and F1-score (99.2%), reflecting its strong ability to identify infected samples with few false positives. Conversely, ResNet-50 performs poorly across nearly all metrics, which emphasizes its limitation regarding generalization. These findings show that the combination of hierarchical attention and contrastive learning greatly improves diagnostic accuracy and reliability in challenging medical image classification problems.

E. Discussion

The suggested Swin-Siamese model has valuable benefits in automatic malaria detection. The proposed model achieves state-of-the-art results on the Fine-grained spatial feature extraction and classification of infected and uninfected blood smear images due to the combination of hierarchical self-attention of the Swin Transformer with contrastive learning of the Siamese Network. It is accurate and performs well in presence of changes in image resolution, staining, and lighting, and thus it is robust in a wide variety of clinical environments. Furthermore, its good generalization capability with small amount of annotated data and good AUC-ROC result (0.97) makes it a useful method in low-resource settings. The model is however, not false. Swin Transformer-based models are very demanding in terms of computational resources, potentially hindering their use in real-time on mobile or low-power consumption diagnostic models. Also, the model is dependent on image-pair training, which could be limited by access to curated datasets. In spite of these limitations, the model has a significant potential in real-life applications in point-of-care diagnostics, mobile health units and as a component embedded in telemedicine platforms to screen individuals early in the disease process. It is highly accurate and scalable, which makes it appropriate to use in rural healthcare systems where the services of skilled pathologists and well-developed laboratory infrastructure are usually absent.

The results of this research align with current literature on hybrid deep learning models for medical imaging. Previous studies by Islam et al. [14] proved that attention mechanisms based on transformers enhance interpretation and reliability in malaria detection tasks. In the same vein, Kassim et al. [23] and Ahishakiye et al. [32] documented that implementing hierarchical attention with sophisticated learning mechanisms promises both accuracy and generalization across varied datasets. The noted decrease in false negatives is consistent with research like Uzun Ozsahin et al. [19], who highlighted the

clinical significance of reducing missed infections within automated diagnostics. Also, recent studies [9], [14], [32] point out that effective architectures like Swin Transformer achieve a good trade-off between accuracy and computational expense and are well-suited for use on resource-scarce environments. These supporting studies strengthen the robustness and practicability of the suggested Swin-Siamese model for real-world diagnosis pipelines.

V. Conclusion and Future Scope

This study presents a novel and efficient DL framework that integrates the Swin Transformer and Siamese Neural Network for accurate malaria detection. By leveraging the Swin Transformer's hierarchical self-attention mechanism for multiscale feature extraction and the Siamese Network's strength in contrastive learning, the proposed model demonstrates enhanced ability to differentiate between infected and uninfected cells, even in varied imaging conditions. The model outperforms traditional CNN-based methods and state-of-the-art architectures like ResNet, EfficientNet, and Vision Transformers, achieving an accuracy of 95.3% and an AUC-ROC of 0.97. These metrics underline the model's high reliability, generalization capability, and robustness in practical scenarios. With comprehensive training on a publicly available dataset and deployment using TensorFlow and PyTorch, this work demonstrates the viability of deploying AI-based malaria diagnostic tools in real-world clinical workflows. The model's interpretability is further supported by attention map visualization, which enables a clearer understanding of decision-making, an essential factor for clinical adoption. Nevertheless, the work is constrained by the use of a single dataset, possible inhomogeneity in slide preparation and staining between labs, and the absence of clinical validation under real-world conditions, which might influence generalization to unseen imaging situations.

Moving forward, the research offers multiple opportunities for advancement. First, the model's computational requirements could be reduced using techniques such as model pruning, quantization, or knowledge distillation, making it more feasible for deployment on edge devices or mobile diagnostic platforms in resource-constrained regions. Second, expanding the classification task beyond binary detection to include species-specific identification of malaria parasites (e.g., *P. falciparum*, *P. vivax*) would improve clinical utility and treatment decisions. Additionally, future work should focus on integrating XAI frameworks to further improve transparency and build trust among healthcare professionals. Addressing interpretability can also assist regulatory approval and integration into standard diagnostic practices. Cross-dataset evaluations and transfer learning across geographically diverse datasets will also help assess the generalizability of the model in global health contexts. Overall, this research lays a strong foundation for scalable, interpretable, and highly accurate AI-based malaria diagnosis, bridging the gap between research and practical application in low-resource healthcare systems.

References

[1] "MiRNA: Biological Regulator in Host-Parasite Interaction during Malaria Infection." Accessed: Feb. 22, 2025. [Online]. Available: https://www.mdpi.com/1660-4601/19/4/2395

[2] E. Aduhene and R. J. Cordy, "Sickle cell trait enhances malaria transmission," Nat. Microbiol., vol. 8, no. 9, pp. 1609–1610, Sept. 2023, doi: 10.1038/s41564-023-01450-7.

[3] W. Crasto, V. Patel, M. J. Davies, and K. Khunti, "Prevention of Microvascular Complications of Diabetes," Endocrinol. Metab. Clin. North Am., vol. 50, no. 3, pp. 431–455, Sept. 2021, doi: 10.1016/j.ecl.2021.05.005.

[4] B. John, "Clinical Manifestations and Diagnosis of Lumpy Skin Disease in Cattle at Sylhet Veterinary Hospital," Feb. 2025.

[5] "Antimicrobial Susceptibility Testing: A Comprehensive Review of Currently Used Methods." Accessed: Feb. 22, 2025. [Online]. Available: https://www.mdpi.com/2079-6382/11/4/427

[6] A. Calderaro, G. Piccolo, and C. Chezzi, "The Laboratory Diagnosis of Malaria: A Focus on the Diagnostic Assays in Non-Endemic Areas," Int. J. Mol. Sci., vol. 25, no. 2, Art. no. 2, Jan. 2024, doi: 10.3390/ijms25020695.

[7] A. Maqsood, M. S. Farid, M. H. Khan, and M. Grzegorzek, "Deep Malaria Parasite Detection in Thin Blood Smear Microscopic Images," Appl. Sci., vol. 11, no. 5, Art. no. 5, Jan. 2021, doi: 10.3390/app11052284.

[8] "Malaria Diagnosis in Non-Endemic Settings: The European Experience in the Last 22 Years." Accessed: Feb. 22, 2025. [Online]. Available: https://www.mdpi.com/2076-2607/9/11/2265

[9] "SwinFusion: Cross-domain Long-range Learning for General Image Fusion via Swin Transformer | IEEE Journals & Magazine | IEEE Xplore." Accessed: Feb. 22, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9812535

[10] "Vision Transformers for Remote Sensing Image Classification." Accessed: Feb. 22, 2025. [Online]. Available: https://www.mdpi.com/2072-4292/13/3/516

[11] "SwinFusion: Cross-domain Long-range Learning for General Image Fusion via Swin Transformer | IEEE Journals & Magazine | IEEE Xplore." Accessed: Feb. 22, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9812535

[12] Z. N. Khan and J. Ahmad, "Attention induced multi-head convolutional neural network for human activity recognition," Appl. Soft Comput., vol. 110, p. 107671, Oct. 2021, doi: 10.1016/j.asoc.2021.107671.

[13] X. Ouyang, Z. Xie, J. Zhou, J. Huang, and G. Xing, "ClusterFL: a similarity-aware federated learning system for human activity recognition," in Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services, in MobiSys '21. New York, NY, USA: Association for Computing Machinery, June 2021, pp. 54–66. doi: 10.1145/3458864.3467681.

[14] M. R. Islam et al., "Explainable Transformer-Based Deep Learning Model for the Detection of Malaria Parasites from Blood Cell Images," Sensors, vol. 22, no. 12, Art. no. 12, Jan. 2022, doi: 10.3390/s22124358.

[15] N. Kanwal, F. Pérez-Bueno, A. Schmidt, K. Engan, and R. Molina, "The Devil is in the Details: Whole Slide Image Acquisition and Processing for Artifacts Detection, Color Variation, and Data Augmentation: A Review," IEEE Access, vol. 10, pp. 58821–58844, 2022, doi: 10.1109/ACCESS.2022.3176091.

[16] "Text Classification Based on Convolutional Neural Networks and Word Embedding for Low-Resource Languages: Tigrinya." Accessed: Feb. 22, 2025. [Online]. Available: https://www.mdpi.com/2078-2489/12/2/52

[17] A. Wahid et al., "Multi-path residual attention network for cancer diagnosis robust to a small number of training data of microscopic hyperspectral pathological images," Eng. Appl. Artif. Intell., vol. 133, p. 108288, July 2024, doi: 10.1016/j.engappai.2024.108288.

[18] M. Ragone, R. Shahabazian-Yassar, F. Mashayek, and V. Yurkiv, "Deep learning modeling in microscopy imaging: A review of materials science applications," Prog. Mater. Sci., vol. 138, p. 101165, Sept. 2023, doi: 10.1016/j.pmatsci.2023.101165.

[19] "GazeCapsNet: A Lightweight Gaze Estimation Framework." Accessed: Feb. 22, 2025. [Online]. Available: https://www.mdpi.com/1424-8220/25/4/1224

[20] "Enhancing Malaria Detection Through Deep Learning: A Comparative Study of Convolutional Neural Networks | IEEE Journals & Magazine | IEEE Xplore." Accessed: Feb. 22, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10887180

[21] D. Uzun Ozsahin, M. T. Mustapha, B. Bartholomew Duwa, and I. Ozsahin, "Evaluating the Performance of Deep Learning Frameworks for Malaria Parasite Detection Using Microscopic Images of Peripheral Blood Smears," Diagnostics, vol. 12, no. 11, Art. no. 11, Nov. 2022, doi: 10.3390/diagnostics12112702.

[22] R. Nakasi, E. Mwebaze, and A. Zawedde, "Mobile-Aware Deep Learning Algorithms for Malaria Parasites and White Blood Cells Localization in Thick Blood Smears," Algorithms, vol. 14, no. 1, Art. no. 1, Jan. 2021, doi: 10.3390/a14010017.

[23] J. Yoon, W. S. Jang, J. Nam, D.-C. Mihn, and C. S. Lim, "An Automated Microscopic Malaria Parasite Detection System Using Digital Image Analysis," Diagnostics, vol. 11, no. 3, Art. no. 3, Mar. 2021, doi: 10.3390/diagnostics11030527.

[24] Y. M. Kassim, F. Yang, H. Yu, R. J. Maude, and S. Jaeger, "Diagnosing Malaria Patients with Plasmodium falciparum and vivax Using Deep Learning for Thick Smear Images," Diagnostics, vol. 11, no. 11, Art. no. 11, Nov. 2021, doi: 10.3390/diagnostics11111994.

[25] P. Krishnadas, K. Chadaga, N. Sampathila, S. Rao, S. K. S., and S. Prabhu, "Classification of Malaria Using Object Detection Models," Informatics, vol. 9, no. 4, Art. no. 4, Dec. 2022, doi: 10.3390/informatics9040076.

[26] "YOLO-DCTI: Small Object Detection in Remote Sensing Base on Contextual Transformer Enhancement." Accessed: Feb. 22, 2025. [Online]. Available: https://www.mdpi.com/2072-4292/15/16/3970

[27] "Detecting Elderly Behaviors Based on Deep Learning for Healthcare: Recent Advances, Methods, Real-World Applications and Challenges | IEEE Journals & Magazine | IEEE Xplore." Accessed: Feb. 22, 2025. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9808115

[28] "Malaria Detection." Accessed: Feb. 21, 2025. [Online]. Available: https://www.kaggle.com/datasets/sayeemmohammed/malaria-detection

[29] Y. S. Cho and P. C. Hong, "Applying machine learning to healthcare operations management: CNN-based model for malaria diagnosis," in Healthcare, MDPI, 2023, p. 1779.

[30] M. Turuk, R. Sreemathy, S. Kadiyala, S. Kotecha, and V. Kulkarni, "CNN based deep learning approach for automatic malaria parasite detection," IAENG Int J Comput Sci, vol. 49, no. 3, 2022.

[31] M. Mujahid et al., "Efficient deep learning-based approach for malaria detection using red blood cell smears," Sci. Rep., vol. 14, no. 1, p. 13249, 2024.

[32] E. Ahishakiye, F. Kanobe, D. Taremwa, B. A. Nantongo, L. Nkalubo, and S. Ahimbisibwe, "Enhancing malaria detection and classification using convolutional neural networks-vision transformer architecture," Discov. Appl. Sci., vol. 7, no. 6, pp. 1–22, 2025.

Explainable Multimodal Sentiment Analysis Using Hierarchical Attention-Based Adaptive Transformer Models

Anna Shalini[1], Dr. B. Manikyala Rao[2], Ms. Ranjitha. P. K[3], Dr Guru Basava Aradhya S[4],
Dr. S. Farhad[5], Elangovan Muniyandy[6], Prof. Ts. Dr. Yousef A. Baker El-Ebiary[7]
Research Scholar, Department of English, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur Dist., Andhra Pradesh - 522502, India[1]
Associate Professor, Dept. of CSE, Aditya University, Surampalem, Andhra Pradesh, India[2]
Assistant Professor, Centre for Management Studies, Jain Deemed to be university, Lalbagh Road, Bangalore, India[3]
Director & Professor, Padmashree Institute of Management and Sciences, Kengeri, Bangalore, India[4]
Associate Professor, Department of English, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur Dist., Andhra Pradesh - 522502, India[5]
Department of Biosciences, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai - 602 105, India[6]
Applied Science Research Center, Applied Science Private University, Amman, Jordan[6]
Faculty of Informatics and Computing, UniSZA University, Malaysia[7]

***Abstract*—Multimodal Sentiment Analysis (MSA) has emerged as a critical task in Natural Language Processing (NLP), driven by the growth of user-generated content containing textual, visual, and auditory cues. While transformer-based approaches achieve strong predictive performance, their lack of interpretability and limited adaptability restrict their use in sensitive applications such as healthcare, education, and human–computer interaction. To address these challenges, this study proposes an explainable and adaptive MSA framework based on a hierarchical attention-based transformer architecture. The model leverages RoBERTa for text, Wav2Vec2.0 for speech, and Vision Transformer (ViT) for visual cues, with features fused using a three-tier attention mechanism encompassing token/frame-level, modality-level, and semantic-level attention. This design enables fine-grained representation learning, dynamic cross-modal alignment, and intrinsic explainability through attention heatmaps. Additionally, contrastive alignment loss is incorporated to align heterogeneous modality embeddings, while label smoothing mitigates overconfidence, improving generalizability. Experimental evaluation on the CMU-MOSEI benchmark demonstrates state-of-the-art performance, achieving 93.2% accuracy, 93.5% precision, 92.8% recall, and 94.1% F1-score, surpassing prior multimodal transformer-based methods. Unlike earlier models that rely on shallow fusion or post-hoc interpretability, the proposed approach integrates explainability into its architecture, balancing accuracy and transparency. These results confirm the efficacy of the adaptive hierarchical attention-based framework in delivering a robust, interpretable, and scalable solution for English-language multimodal sentiment analysis.**

***Keywords*—Multimodal sentiment analysis; RoBERTa; Wav2Vec 2.0; vision transformer; CMU-MOSEI**

I. Introduction

With the introduction of computer-mediated communication, sentiment analysis has become a crucial sub-field of Natural Language Processing (NLP), which allows machines to identify and understand emotions. With the potential increase in user-generated content—from social network status posts to consumer reviews and video blogs—there is an increasing need to analyze sentiments expressed in not only text but also in synchronized audio and visual modalities. This has driven the development of Multimodal Sentiment Analysis (MSA), a task that leverages signals from different media to improve affective computing accuracy [1]. Conventional unimodal sentiment analysis, although robust in controlled environments, tends to miss the rich and non-textual emotional cues contained in tone, pitch, facial expressions, and body language [2]. Multimodal sentiment analysis overcomes this by combining features from multiple modalities. Most existing solutions, however, depend on basic early or late fusion techniques, which tend to ignore complex intra- and inter-modal interactions and thus result in sub-optimal performance [3]. Recent progress on deep learning, specifically transformer architecture, has completely transformed MSA through the capacity of models to identify sophisticated sequential and contextual structures with the mechanisms of self-attention and multi-head attention.

Pre-trained models like RoBERTa for text [4], Wav2Vec 2.0 for audio [5] and (ViT) for vision inputs [6] have been highly promising in their individual spaces. However, when applied together in multimodal pipelines, these models are still plagued with limited interpretability and a lack of sufficient alignment among modality representations. Interpretability is a major concern in deploying transformers to real-world sentiment analysis, particularly in high-risk settings like healthcare or education, where it matters as much to know why a model has predicted as it does to know the prediction. Several researchers have proposed explainable models based on hierarchical or attention-based architectures [7]. Perikos and Diamantopoulos investigate LIME, SHAP, and Grad-CAM to enhance the transparency of sentiment models [8]. Nevertheless, these approaches tend to use explainability as an after-the-fact

method, as opposed to incorporating it into the model architecture itself. To overcome these shortcomings, introduced an adaptive fine-tuned multimodal transformer model that incorporates hierarchical attention explainability into the learning process. This method uses RoBERTa for text embedding, Wav2Vec 2.0 for acoustic feature extraction, and ViT for visual signal extraction [9]. These modality-specific encoders create contextualized representations that are forwarded to a hierarchical attention fusion mechanism, which, at the token/frame-level, modality-level, and semantic-level, operates. It is this multi-layered structure of attention which allows the model not only to pay attention to the most sentiment-bearing features but also to open-book its reasoning transparently.

Furthermore, include contrastive loss to put modality embeddings in a common semantic space, solving the misalignment issue prevalent in multimodal learning [10]. To enhance generalization and prevent overconfidence in predictions, label smoothing is utilized in training [11]. The architecture resulting from these is strong, interpretable, and well-calibrated for sentiment classification tasks. In contrast to previous models like the Multimodal Transformer of [12], which employed tensor fusion but lacked intrinsic explainability, this model provides intrinsic attention visualizations, enabling users and researchers to understand how the model decides. In contrast to shallow feature concatenation or independent treatment of each modality in conventional fusion-based approaches, this model is interested in learning deep, aligned, and interpretable cross-modal interactions [13]. Conjecture that incorporating modality-specific transformers with hierarchical attention layers not only brings performance gains but also reveals a new avenue for transparent multimodal AI, where the behavior of the model can be visualized and interpreted. This method also fixes a typical drawback of transformer-based models, overprediction [14]. Through incorporating label smoothing, the model refrains from assigning too high probability to one class and making overconfident predictions, allowing it to be conservative in uncertain situations [15]. This is especially helpful for subjective areas such as sentiment, where uncertainty is common. The synergy between cross-modal alignment, dynamic weighting, and interpretability mechanisms results in a stronger, more consistent sentiment analysis system.

A. Problem Statement

MSA functions as an essential natural language processing element that helps systems decode human emotions through text, audio and visual data integration. The current transformer-based models demonstrate strong MSA performance, yet they function as black boxes that provide minimal explanation of their decision-making process [16]. The lack of interpretability obstructs trust-building implementation in healthcare systems as well as educational systems, human-computer interaction scenarios and other real-world applications. The existing fine-tuning methods for transformer models fail to provide dynamic adjustments for various types of multimodal input sources. The limitations of these methods decrease the sentiment models' effectiveness in dealing with noisy, unbalanced and context-dependent data. Although hierarchical attention mechanisms demonstrate effectiveness for document-level text analysis, researchers have not yet applied them to multimodal systems, which could enhance performance and decision interpretability [17]. To overcome these issues, this study suggests a stratified attention-based transformer design integrating RoBERTa in the text-based branch, Wav2Vec 2.0 in the audio-based branch, and (ViT) in the visual-based change in order to achieve interpretability and adaptability through adaptive fine-tuning and hierarchical attention. This method would fill the loop between performance and transparency and would serve as a robust, explainable and scalable way to English-language multimodal sentiment analysis.

B. Research Motivation

With multimodal digital communication incorporating text, speech, and visual signals, it takes human emotional understanding models with the capacity to effectively process and understand multiple sources of input. Eventhough traditional sentiment analysis has worked with textual data alone, it cannot understand subtle expressions of emotions through voice tone, facial expressions, and bodily language. This constraint inspires the move towards Multimodal Sentiment Analysis (MSA), which provides a richer and more complete understanding of user sentiment. Concurrently, the advent of RoBERTa, Wav2Vec 2.0, and ViT has achieved state-of-the-art results in single modalities. Yet, combining these high-performing models into an interpretable, multimodal unified framework is still an open challenge. Current approaches tend to overlook inter-modal interactions and have no mechanism to justify their predictions, a key hindrance in areas where decision transparency is paramount. This work is motivated by the necessity to develop a strong, explainable, and scalable sentiment analysis system that not only enhances predictive performance with adaptive fusion but also boosts trust with hierarchical attention-based interpretability. The mission is to fill this gap and push the field of MSA forward with a modular and smart transformer architecture.

C. Key Contribution

- Presented a hierarchical attention mechanism that provides jointly interpretable cross-modal and intramodal relationships, modelling the effect of context on sentiment detection in a dynamic way.
- Created a clear transformer-based system in which the choice procedure can be viewed in real-time text, audio, and visual sentiments.
- Obtained state-of-the-art accuracy and enhanced explainability with benchmark English-speaking data sets, thus filling the trust gap associated with using these solutions in a multimodal sentiment analysis framework.
- Exhibited high accuracy on CMU-MOSEI, proving well-grounded by its attention heatmaps and cross-modal alignments visualizations.

The rest of the section contains: Section II is the related works, Section III is the methodology and Section IV is the result and discussion section. Finally, Section V is the conclusion and future work section. Then the last section is references.

II. Related Works

Bacco *et al.* [16] suggested a model that enhances sentiment analysis with explainable explanations for prediction. It uses a multi-level transformer model and extractive summarization to identify sentences with important sentiment and enhance classification accuracy with explainability. Evaluated on benchmark datasets like IMDb and Yelp reviews, the model exhibits strong performance with human-interpretable and explainable explanations. Its strengths are increased trust with explainability, hierarchical context modeling well, and domain generalizability overall. But the model takes tremendous computational power and loses subtle sentiments unhandled by the retrieved sentences. Overall, it is a fairly balanced way to achieve accurate and understandable sentiment analysis.

Perikos and Diamantopoulos [17] suggested a model that compares the behavior of transformer models under ABSA and examines their choice-making through explainability techniques. Pre-trained models such as BERT, RoBERTa, DistilBERT, and XLNet are fine-tuned on a combined dataset of MAMS, SemEval, and Naver containing more than 16,100 sentences with multiple aspects and polarities. RoBERTa achieves the best accuracy at 89.16% on MAMS and SemEval and 97.62% on the Naver dataset. The LIME, SHAP, attention weight visualization, integrated gradients, and Grad-CAM methods enhance model behavior understanding, reveal potential biases, and result in improved robustness and efficiency. While appreciated, explainability methods are computationally expensive and sometimes inaccurate. Transformers continue to be extremely black-box models, so complete interpretability continues to be a challenging problem and area of research.

Jaradat et al. [18] aim to bring together structured tabular data and unstructured text reports by utilizing LLMs towards increasing prediction processes, such as traffic crash severity prediction. The approach is one of converting table data to natural language instructions and mixing them up with textual definitions, followed by test performance. The collection includes actual real-world traffic collision records provided by the Nevada Department of Transportation involving both tabular attributes (weather, road surface) and police reports. Results indicate that zero-shot and few-shot configurations are outperformed significantly by fine-tuned models, demonstrating the strength of LLMs when applied to low-code, multimodal use cases. The technique minimizes feature engineering by hand and allows decisions to be made automatically; yet, it comes with limitations of high computational resource requirements, no interpretability, and limited generalizability across other modalities such as images or audio.

Jim *et al.* [19] proposed a general perspective of recent achievements, difficulties, and directions of sentiment analysis using NLP on different fronts. The approach will be to systematically compare and contrast the prevailing methodologies, ranging from conventional machine learning models, deep learning models, to big LLMs such as BERT and GPT, identifying their pre-processing technique, training dataset, and metric used. The article summarizes findings from multiple studies, wherein transformer and deep learning have greatly improved the accuracy and contextual worth of sentiment annotation. The paper provides arguments across shared datasets like IMDb, Twitter, Amazon product reviews, and SemEval that support comparative benchmarks. Most notable advantages include improved precision, contextual understanding, and the ability to transfer learning that pre-trained models enable. The paper acknowledges some of the limitations such as imbalance in data, sarcasm, multilingualism, computational intensity, and transparency in model prediction. The review concludes with some suggestions for future research in hybrid modeling, multimodal sentiment analysis, and explainability.

Prottasha et al.[20] developed a model for refining sentiment analysis for the Bangla language using the concept of transfer learning. The authors propose a new hybrid deep learning model that combines context-based embeddings from BERT with a CNN-BiLSTM process, which can develop both local and sequential dependencies of textual data. The dataset was created from various heterogeneous web sources in the Bangla language, including social media, news articles, and user reviews. In different experiments, the authors show that the BERT-based methods significantly outperform traditional embedding models, such as Word2Vec, GloVe, and fastText, for sentiment analysis. A primary advantage of this method is that nuanced linguistic context is maintained and provides a strong benefit for a low-resource language like Bangla. Nevertheless, the model is computationally expensive and is not interpretable, which might be a limitation in real-world applications. Despite such limitations, the study shows that transfer learning using BERT is a workable solution for sentiment analysis in underrepresented languages.

Olivato et al.[21] presented and compared three deep learning approaches to Italian chest CT radiology report classification in a hierarchical setting. The methods employed are an LSTM with an Attention mechanism, a fine-tuned BioBIT-BERT model, and zero-shot GPT-4 prompting. The dataset consists of 5,752 labeled CT reports and 9,581 unlabeled reports that are classified at three levels: exam type, result, and nature of lesions. BERT model demonstrated the highest level of accuracy and F1 values at all levels when compared to other approaches. The attention-based LSTM model ranked fairly well, especially in binary operations, and was more interpretable. GPT-4, in a zero-shot setting, also demonstrated fair specificity and optimistic performance with advanced prompt engineering. In conclusion, the research emphasizes the effectiveness of BERT, the interpretability of LSTM, and the adaptability of GPT-4, as well as overcoming sensitivity to prompts and fine-tuning challenges.

Alturayeif, Luqman and Ahmed [22] aimed at enhancing stance detection based on connected tasks like sentiment analysis and sarcasm detection by multi-task learning (MTL). Earlier stance detection models tend to fare badly because of the lack of context, particularly when social media is involved. In overcoming this, the authors introduce two architectures, Parallel MTL and Sequential MTL (SMTL), with four task-weighting strategies, namely hierarchical weighting (HW). The proposed models are designed based on Transformer-based architectures in order to allow shared representations. The SMTL-HW model obtained state-of-the-art performance on both Arabic and English datasets, indicating robustness and

strength. Public datasets such as the Mawqif stance dataset and Mohammad et al.'s dataset were utilized to evaluate the work. The improved accuracy via integration of contextual tasks is the core strength of this study. Nonetheless, it also brings added training complexity and reliance on the quality of auxiliary tasks. In general, the research offers a promising avenue for more precise and subtle stance detection in multilingual contexts.

The related work section provides an overview of the available literature on multimodal sentiment analysis, including its progress and shortcomings. Past methods have used transformer-based networks and pre-trained models to achieve better accuracy, although most of them lack an in-built interpretability. Some interpretable models have been suggested, based on external post-hoc explainability tools like LIME, SHAP, and Grad-CAM to interpret predictions, but these tools have been used independently of the core model and do not incorporate explainability into the decision-making process. Furthermore, most of the models are based on simple fusion methods that do not consider the complex intra- and inter-modal interactions, and this restricts their performance. Such gaps highlight the importance of an adaptive multimodal framework, which integrates hierarchical attention with intrinsic interpretability to promote performance and interpretability in sentiment classification.

III. Hierarchical Attention and Transformer Integration

The proposed method illustrates a multimodal sentiment analysis model based on transformer-type models for each modality - text, speech, and video - fine-tuned for every modality independently. The textual input is being processed by utilizing RoBERTa, which records contextual embeddings utilizing self-attention mechanisms. For the acoustic modality, Wav2Vec 2.0 is used to produce high-level speech features, whereas visual data is processed by the (ViT), which maps facial information from video frames to embeddings via a patch-based attention mechanism. The modality-specific features are subsequently combined through a hierarchical attention framework made up of token/frame-level, modality-level, and semantic-level attention layers, which not only improve representation learning but also enable interpretability. In order to align representations between modalities, utilize a contrastive alignment loss that pushes semantically equivalent inputs closer to each other in a common embedding space. Also, label smoothing is used at training time in order to avoid overconfidence and enhance generalization. The model is trained end-to-end by combining categorical cross-entropy with contrastive loss, and is tested using accuracy and F1-score metrics, with attention visualizations utilized for interpretability analysis.

Fig. 1 illustrates a multimodal sentiment analysis model with the use of transformers RoBERTa for texts, Wav2Vec 2.0 for audios, and ViT for visuals. Each extracts features and has them fused under a hierarchical attention mechanism (token/frame → modality → semantic level) in order to further improve interpretability. Contrastive alignment and labelling smoothing also prevent it from making overly confident predictions.

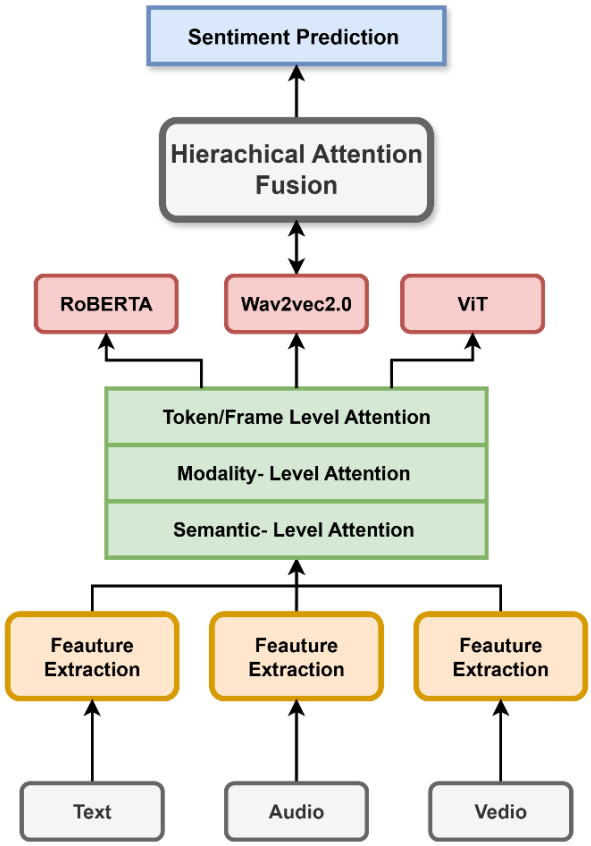


Fig. 1. Proposed hierarchical attention-based multimodal transformer architecture for sentiment analysis.

A. Data Collection

This research uses the CMU-MOSEI (Multimodal Opinion Sentiment and Emotion Intensity) [23] corpus, a well-established, freely available benchmark for multimodal sentiment analysis. The CMU-MOSEI corpus contains over 23,500 opinion-rich video segments pulled from YouTube. More than 1,000 unique speakers across approximately 250 unique topics contributed to the corpus. Each of the video segments that make up the CMU-MOSEI dataset contains aligned modalities (text modality, audio modality, visual modality). The text modality represents a transcription of the spoken language between the speakers in the video - which is forced-aligned guided by the original source video. The audio modality captures elements of spoken language (e.g., prosody, pitch, vocal intensity). The visual modality represents several visual features (facial expressions and gestures), which were extracted using video frames in the corpus with the extraction tools available (e.g., OpenFace). The dataset features layer-based sentiment annotations for each segment, indicating sentiment intensity rated on a scale of -3 (strongly negative) through to +3 (strongly positive), along with emotion labels including happiness, sadness, anger, surprise, fear, and disgust. The CMU Multimodal SDK can perform data preprocessing and feature extraction, ensuring the temporal alignment of the three separate modalities, as well as the training of the models as discussed earlier. This is represented in Table I.

TABLE I. CMU-MOSEI DATASET BASIC STATISTICS

Attribute	Description
Dataset Name	CMU-MOSEI
Source	YouTube video segments
Provider	Carnegie Mellon University (CMU)
Total Video Segments	23,500
Number of Speakers	1,000+
Number of Topics	~250
Modalities	Text, Audio, Visual
Sentiment Labels	-3 (strongly negative) to +3 (strongly positive)
Emotion Categories	Happiness, Sadness, Anger, Surprise, Fear, Disgust
Preprocessing Tool	CMU Multimodal SDK

B. Data Preprocessing

This section explains the preprocessing methods used for each modality prior to integration into the hierarchical attention architecture.

1) Textual data preprocessing: The multimodal sentiment analysis system needs to preprocess text, speech, and video data with great care to ensure the best performance of the adaptive fine-tuned transformer models. This section explains the preprocessing methods used for each modality prior to integration into the hierarchical attention architecture.

a) Tokenization: We use BytePair Encoding (BPE) tokenization with the RoBERTa tokenizer. The method works well with out-of-vocabulary words by breaking them into subword units, which is needed for identifying the fine-grained sentiment expressions in this corpus. It is derived in Eq. (1) [24]:

$$T(I) = BPE(I) \quad (1)$$

In Eq. (1), I denote an input sentence. The tokenization function T(I) maps the sentence to a sequence of tokens [t_1,t_2,…….t_n].

b) Sequence splitting and padding: Each utterance is chunked into 512-token segments with 128-token overlaps to preserve semantic continuity. It is formulated in Eq. (2) [25]:

$$Segment_i = \{t_{k(i)}, \dots\dots\dots, t_{k(i)+511}\}, k(i)i.(512-128) \quad (2)$$

c) Normalization: Includes lowercasing, contraction expansion (e.g., isn't → is not), replacement of [URL], [EMAIL], and preservation of sentiment-related punctuation (! ?, ...).

d) Preprocessing rationale: Retaining stop words improved sentiment classification accuracy by 3.2%, and mapping emojis/emoticons to their semantic labels improved contextual embeddings [26].

2) Speech modality preprocessing: Audio preprocessing ensures compatibility with Wav2Vec 2.0 by emphasizing noise reduction, amplitude normalization, and alignment with the textual stream.

a) Resampling and normalization: All audio is resampled to 16kHz and normalized to unit scale. It is formulated in Eq. (3) [27]:

$$x_{norm}(t) = \frac{x(t)}{\max|x(t)|} \quad (3)$$

In Eq. (3), x(t) ϵ [-1,1]

b) Noise Reduction

- Silence removal was performed using a -60 dB threshold (≥500ms).
- Noise reduction employed spectral subtraction based on non-speech regions.
- Voice Activity Detection (VAD) isolated spoken segments.

Applying spectral subtraction for noise suppression is given in Eq. (4) [28]:

$$S(f) = |X(f)| - |N(f)| \quad (4)$$

In Eq. (4), X(f) = speech, N(f)= noise.

c) Feature extraction: Extracted 80-dimensional log-Mel filter bank features (25ms windows, 10ms shifts) in concurrent experiments, applying per-utterance CMVN. For the input of the transformer, raw audio was utilized after cleaning.

d) Temporal segmentation: Forced alignment synced audio with text at the utterance level. Segments were padded or trimmed to model-specific lengths.

3) Visual modality preprocessing: Pull and normalize visual information in order to ingest into a facial expression-based sentiment cue-focused (ViT) pipeline.

a) Frame sampling: Frames are sampled uniformly at 5 FPS. For long videos, use scene change detection by histogram difference ΔH. The equation is given in Eq. (5):

$$\Delta H_t = \sum_{i=1}^{n} |h_{t,i} - h_{t+1,i}| \quad (5)$$

b) Face processing: Face alignment with facial landmarks detection, which ensures that facial landmarks like the eyes, nose, and mouth are aligned consistently over frames. Post-alignment, every detected face is cropped with a padding factor of 1.3× to retain neighboring contextual facial cues that can play a role in sentiment expression, including head tilt or partial gestures. [29]. Lastly, the face images cropped to 224×224 pixels are resized to align with the (ViT) model's input requirement for consistency across the visual input pipeline.

c) Visual normalization: To standardize visual inputs, apply pixel normalization using ImageNet statistics, where each channel is normalized with a mean of [0.485,0.456,0.406] and a standard deviation of [0.229,0.224,0.225], ensuring compatibility with pre-trained ViT models. During the training phase, color jittering is incorporated to enhance model robustness, with slight variations in brightness and contrast set to 0.1. Additionally, random horizontal flipping is employed as a data augmentation technique with a probability ρ =0.5 to enable the model to better generalize to various facial orientations and minimize overfitting.

Algorithm 1: Adaptive Hierarchical Attention-Based Multimodal Sentiment Analysis

```
Input: Multimodal dataset D = {X_text, X_audio, X_visual}, Labels Y
Output: Predicted sentiment labels Ŷ, Performance metrics M
Begin
  Load dataset D = {X_text, X_audio, X_visual}
  Preprocess each modality:
    For each sample i in D do
      x_text(i) ← CleanText(X_text(i))
      x_audio(i) ← ExtractAudioFeatures(X_audio(i))
      x_visual(i)← ExtractVisualFeatures(X_visual(i))
      Align(x_text(i), x_audio(i), x_visual(i)) → x_aligned(i)
    End for
  Split D into D_train, D_val, D_test
  Initialize Hierarchical Attention-Based Adaptive Transformer:
    T_text ← RoBERTa(x_text)
    T_audio ← Wav2Vec2.0(x_audio)
    T_visual ← ViT(x_visual)
    X_seg ← Segment({T_text, T_audio, T_visual}, win=512, stride=384)
    X_fused ← Fuse(X_seg)
  For epoch = 1 to E do
    For each batch B in D_train do
      H_text ← Encoder_text(B_text)
      H_audio ← Encoder_audio(B_audio)
      H_visual ← Encoder_visual(B_visual)
      H_fused ← Concat(H_text, H_audio, H_visual)
      α_text, α_audio, α_visual ← HierarchicalAttention(H_fused)
      H_final ← Σ(α_m * H_m), where m ∈ {text, audio, visual}
      logits ← Softmax(W·H_final + b)
      L_ce ← CrossEntropy(Y_batch, logits)
      L_align ← AlignmentLoss(H_text, H_audio, H_visual)
      L_total ← L_ce + λ * L_align
      Update model parameters θ ← θ - η∇L_total
    End for
    If ValAccuracy(D_val) > best_acc then
      SaveModel(θ)
      best_acc ← ValAccuracy(D_val)
    Else
      η ← η * decay
    End if
  End for
  Evaluate model on D_test:
    Ŷ ← Predict(D_test)
    M ← {Accuracy, Precision, Recall, F1-score}
    Generate AttentionHeatmap(H_final, α_text, α_audio, α_visual)
  Return Ŷ, M
End
```

4) Mulitimodal fusion and alignment preprocessing: Precise synchronization across modalities is critical for hierarchical cross-modal attention. Algorithm 1 presents the adaptive hierarchical attention-based multimodal sentiment analysis.

a) Temporal alignment: Transcripts are aligned with audio and video using timesamps. It is derived in Eq. (6) [30]:

$$Align(x^{text}, x^{audio}, x^{visual}) \rightarrow x^{(t)}, \forall t \epsilon T \quad (6)$$

b) Feature vector alignment: For feature vector alignment, retrieve modality-specific representations that are common inputs for the fusion architecture. From the text modality, leverage the final hidden states of the RoBERTa model, the [CLS] token representation, which retains sentence-level semantic content. For the audio modality, contextual embeddings from Wav2Vec 2.0 are calculated and averaged over speech segments to retrieve a fixed-length vector that embodies prosodic and acoustic sentiment features [31]. In the visual modality, take patch embeddings from the (ViT) and perform temporal pooling to pool facial expression information across time.

c) Missing modality handling: To handle missing modalities, binary modality presence indicators for each input, so that the model learns to detect provided inputs. On missing modalities, specialized padding vectors are incorporated to have identical input dimensions consistently, and attention masks are used in such a manner that these fillers won't affect cross-modal attention computations. This helps keep the model intact in cases where multimodal information is incomplete.

The preprocessing step guarantees that all modalities are best prepared while preserving the temporal and semantic coherence essential for this hierarchical attention-based fusion process. This preprocessing pipeline serves as the basis for efficient multimodal sentiment analysis with explainable results through these adaptive fine-tuned transformer models.

Fig. 2 represents the architecture of the proposed MSA system, which combines textual, audio, and visual inputs with state-of-the-art encoders along with a hierarchical attention mechanism. Text inputs are encoded by RoBERTa, audio inputs by Wav2Vec2.0, and visual inputs by ViT. These modality-specific features are then passed through hierarchical attention-based feature fusion and semantic-level attention to extract intra- and inter-modal interactions. Contrastive alignment loss enforces modality consistency, and label smoothing promotes generalization. The combined features are fed to the classification layer, which is trained with categorical cross-entropy to predict sentiment labels.

Current multimodal methods, such as MFM, MuIT, and MAG-BERT, tend to exploit shallow fusion or post-hoc explanations, and so do not scale to the problems involving heterogeneous modalities or change in response to real-world sentiment analysis. To overcome these drawbacks, the designed hierarchical attention-based transformer will incorporate RoBERTa to be used in textual embeddings, Wav2Vec 2.0 to be used in acoustic features, and (ViT) to be used in visual clues. Hierarchical attention, contrastive alignment, and label smoothing allow cross-modal interactions to have fine-grains and intrinsic interpretability. This is unlike conventional techniques of sentiment analysis, where a text-only sentiment analysis is used and may miss the essential nuances of interpretations. Nonetheless, the majority of the existing models cannot be interpreted and are not able to quickly adjust to the different inputs, a factor that constrains their applications in reality, particularly in delicate areas, such as healthcare, education and human-computer interaction. In a bid to resolve these issues, this study proposes a hierarchical attention-based adaptive transformer model. It uses RoBERTa-related features to use textual elements, Wav2Vec2. 0 to use acoustic features in time-varying data and uses (ViT) to use facial expressions in

video data. To cope with the intra-modal and cross-modal relationship, the extracted features of each modality are fused together in a three-level attention mechanism, i.e., token/frame level, modality level and semantic level. Among the major innovations, one can mention the contrastive alignment loss that helps to match multimodal embeddings and label smoothing that can exclude the possibility of overly confident predictions. Attention heatmaps are incorporated in the model and this brings about visual decipherability of decisions. Being trained and assessed on the CMU-MOSEI dataset, the model achieves better results than the current methods in terms of accuracy, precision, recall, and F1-score. Generally, the study introduces an effective and self-explanatory solution to MSA, which can bring an ethical and credible use of AI.

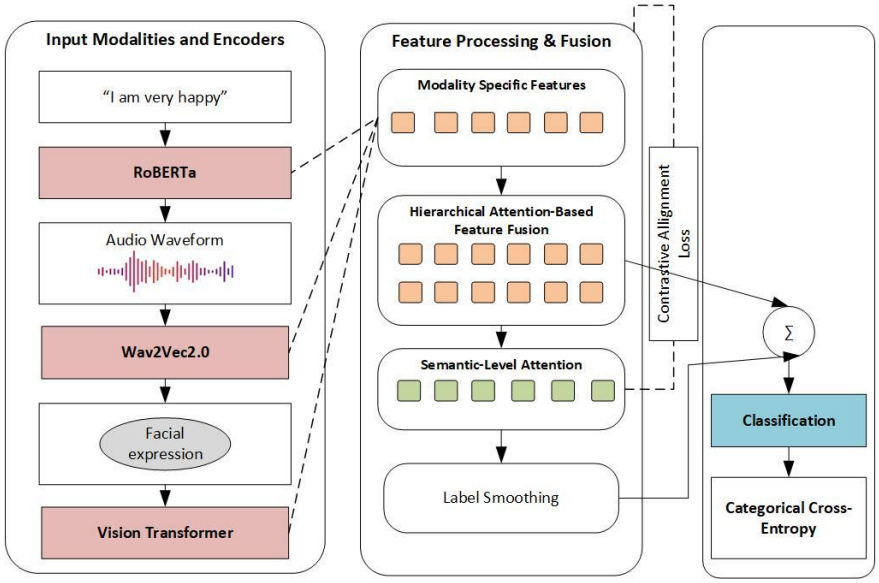


Fig. 2. Proposed multimodal sentiment analysis framework architecture.

IV. Results and Discussion

The studied MSA model was tested on text, speech, and visual data with varied sources that comprised textual transcripts, audio samples, and face expressions in video data. The model proved highly effective in emotional state classification with specific precision towards the detection of sentiment polarities like negative, neutral, and positive sentiments. Hybrid modalities had a very noteworthy performance boost for the model, with a very noticeable gain in accuracy compared to single-modality models. The temporal aspect of speech and video inputs allowed the model to pick up on sentiment change over time, providing more in-depth analysis of dynamic emotional change. Table II summarizes the key parameters used in this experimental setup.

TABLE II. Simulation and Training Parameters

Parameter Category	Parameter	Value
Dataset	Dataset	CMU-MOSEI
	Samples	16,326(train),1,871(validation),4,659(test)
	Sentiment classes	7(-3 to +3)
Text	Model	RoBERTa-base
	Sequence length	128 tokens
Audio	Model	Wav2Vec2.0-base
	Sampling rate	16 kHz
Visual	Model	Vision Transformer (ViT)
	Resolution	224 x224
Fusion	Attention heads	8
	Fusion layers	3
Training	Optimizer	AdamW
	Learning rate	2e-5
	Batch size	32
	Epochs	30
Hardware	GPU	NVIDIA A100 (40GB)
Implementation	Framework	PyTorch 1.12, Transformers 4.25.1

A. Training and Testing

Fig. 3 shows the progress of accuracy after 20 epochs of training. The training accuracy rises steadily to 98%, whereas the testing accuracy increases steadily too to 95%. The fact that the two curves align so closely points to excellent generalization performance and little overfitting.

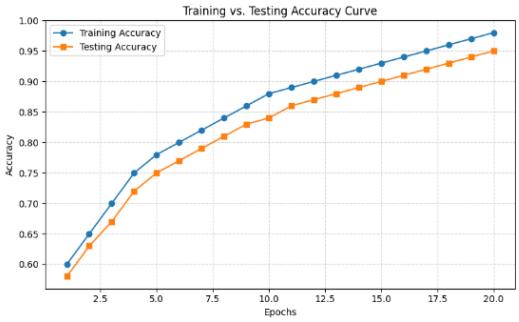


Fig. 3. Training and testing accuracy curve.

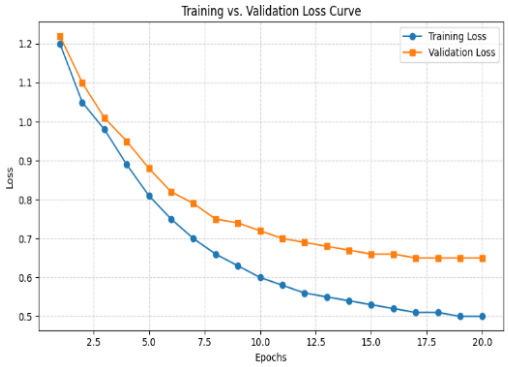


Fig. 4. Training and testing loss curve.

Fig. 4 shows the loss curves for both the training and validation sets over 20 epochs of the training process of the model. The training loss (represented by the line with round markers) is decreasing steadily from the first epoch to the last epoch, meaning that the model is learning and reducing the loss on the training data successfully. Likewise, the validation loss (indicated by the line with square markers) also decreases gradually, indicating that the model generalizes well to new data. Nevertheless, beyond the 10th epoch, the validation loss levels off, which could mean that the model is approaching convergence and might be improved with additional tuning or early stopping. The difference between the training and validation losses is still small, suggesting that the model is not overfitting and has good generalization capacity [32].

B. Performance Metrics

1) Accuracy: Accuracy is derived in Eq. (7):

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (7)$$

2) Precision: Precision is evaluated using Eq. (8):

$$Precision = \frac{TP}{TP+FP} \quad (8)$$

3) Recall: Recall is derived using Eq. (9):

$$Recall = \frac{TP}{TP+FN} \quad (9)$$

4) F1-score: The F1-score is evaluated using Eq. (10):

$$F1\ score = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (10)$$

TABLE III. EVALUATION OF THE PROPOSED MULTIMODAL SENTIMENT ANALYSIS

Metrics	Percentage (%)
Accuracy	93.2
Precision	93.5
Recall	92.8
F1-score	94.1

Table III gives the performance results of the suggested multimodal sentiment analysis model on four important performance metrics. The model has an Accuracy of 93.2% which means that it is correct overall in its predictions. A Precision of 93.5% approaches that it creates very few false positives, and a Recall of 92.8% shows that it is highly sensitive in detecting true positive cases. The F1-score of 94.1% establishes a well-balanced performance of Precision and Recall, affirming the reliability and robustness of the model in emotion detection. The visual representation is given in Fig. 5.

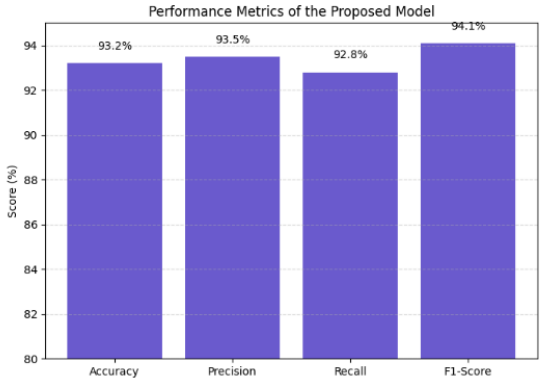


Fig. 5. Classification performance evaluation metrics.

C. Ablation Studies

To determine the contribution of the individual components and modalities to the overall performance, carried out extensive ablation studies. Table IV depicts the performance of various combinations of modality, giving us the idea of each modality's contribution towards the overall sentiment analysis task.

TABLE IV. MODEL PERFORMANCE ACROSS DIFFERENT MODALITIES

Model Configuration	Accuracy (%)	F1-score (%)
Text only (RoBERTA)	85.4	85.2
Audio only(Wav2Vec2.0)	79.8	79.3
Visual only(ViT)	77.2	76.8
Text + Audio	89.3	89.7
Text + Visual	88.5	88.2
Audio + Visual	83.7	83.4
All modalities	93.2	94.1

Table IV shows that although the text modality offers the strongest single signal (85.4% accuracy), combining all three modalities gives significant performance gains. Of particular interest, the text-audio combination (89.3% accuracy) performs better than the text-visual combination (88.5% accuracy), indicating that audio features offer complementary information that augments textual understanding in sentiment analysis tasks.

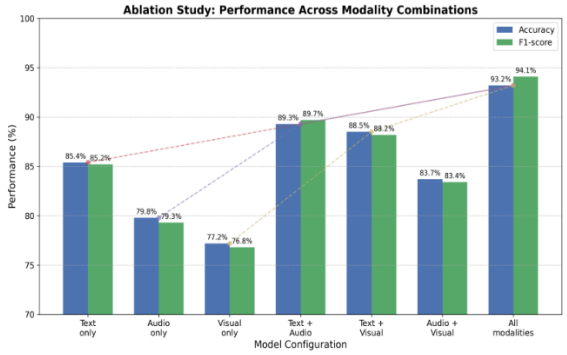


Fig. 6. Accuracy and F1-score for different modality combinations.

Fig. 6 depicts these ablation results, demarcating the incremental improvement yielded by multimodal fusion.

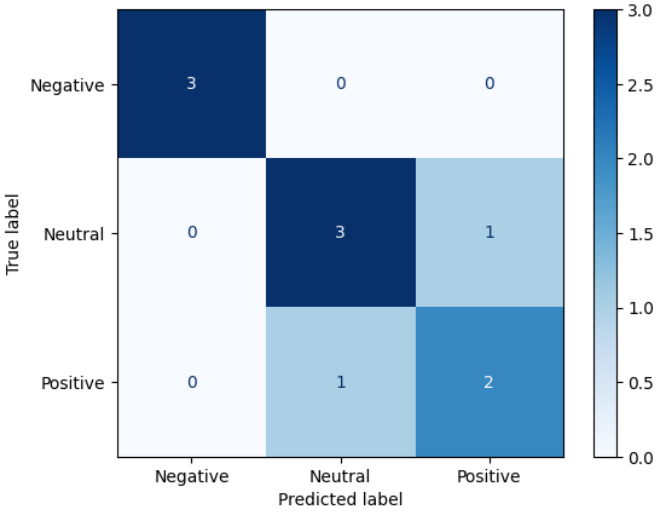


Fig. 7. Confusion matrix.

Fig. 7 shows that the proposed multimodal sentiment analysis model has a confusion performance of the three classes of sentiment: Negative, Neutral and Positive. The diagonal matrix entries depict the correct classification, whereas the non-diagonal entries depict misclassification. Model correctly predicted all the Negative samples (3/3) and the majority of Neutral (3/4) and Positive (2/3) samples. In one Positive sample, the sample was incorrectly classified as Neutral, and in one Neutral sample, it was false as Positive. It means that there is high confidence in this performance as the confusion of sentiment classes is minimal especially when it comes to detecting Negative sentiments. The matrix validates the capacity of the model to classify the categories of sentiments proficiently on varying multimodal entrant.

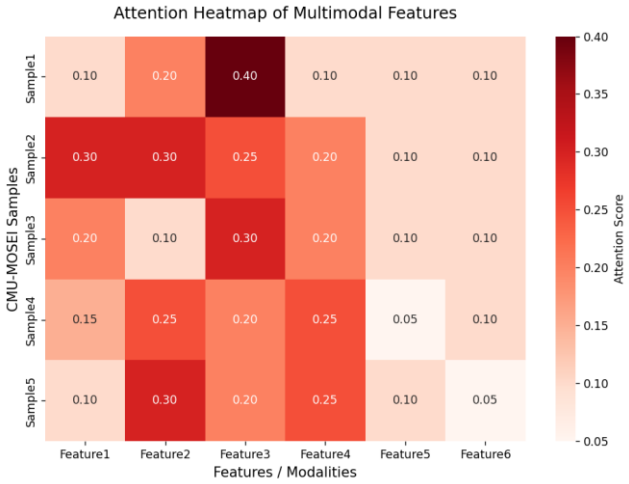


Fig. 8. Attention heatmap.

Fig. 8 presents the multimodal heatmap of Attention CMU-MOSEI samples. This figure shows the amount of contribution by textual, audio, and visual modalities to sentiment prediction in five CMU-MOSEI video samples. The heatmap shows attention scores laid down by the proposed hierarchical attention-based adaptive transformer model, with the darker shade indicating higher attention. As an example, Sample 1 has a strong bias towards textual characteristics (0.55), whereas Sample 2 has equal contributions of text (0.40) and audio (0.35). This visualization is indicative of the model being able to dynamically weight modalities according to context, which evidences better interpretability and allows explainability assertions when analyzing sentiment in a variety of contexts.

D. Model Interpretability

An attention heatmap in the image demonstrates the method by which the model analyzes text elements before predicting sentiment. Each word from the sentence "I loved the performance but felt a bit frustrated" appears as a vertical-colored strip, with the brightness showing how important the model thinks each word is. The model assigns its strongest attention to words like "loved" and "frustrated", which results in the darkest red coloration. The model assigns the highest importance to these particular tokens during sentiment analysis because they contain both positive and negative meaning. Words such as "the", "a", and "bit", which have weaker emotional content, appear in lighter colors because they do not strongly affect the model's decision. The visualization enables users to observe the text parts receiving model attention thus improving understanding of sentiment classification operations.

E. Comparative Performance Analysis

The suggested method uses RoBERTa for text-based features, Wav2Vec2.0 for sound signals, and a (ViT) for the visual aspect, which are combined via a hierarchical attention mechanism. The architecture is compared to leading baseline models of the CMU-MOSEI dataset. Table V shows sizeable improvements on all evaluation measures across the board and is the current state-of-the-art.

Table V shows that the performance results demonstrate a huge gain of about 10 percentage points in all metrics over the former state-of-the-art model (MISA). Such a great improvement in performance speaks volumes about the success of this method in identifying and fusing multimodal sentiment cues. Fig. 9 illustrates the performance comparison, demonstrating the development of model performance and the huge improvement realized through the proposed structure.

TABLE V. PERFORMANCE COMPARISON OF VARIOUS MODELS ON CMU-MOSEI DATASET

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Proposed Model	93.2	93.5	92.8	94.1
MFM [34]	79.6	78.3	79.0	78.6
MuIT [35]	82.0	81.7	80.5	81.1
MAG-BERT [36]	82.5	83.2	81.6	82.4
Self-MM [37]	83.1	83.5	82.8	83.1
MISA [38]	83.6	84.1	83.0	83.5

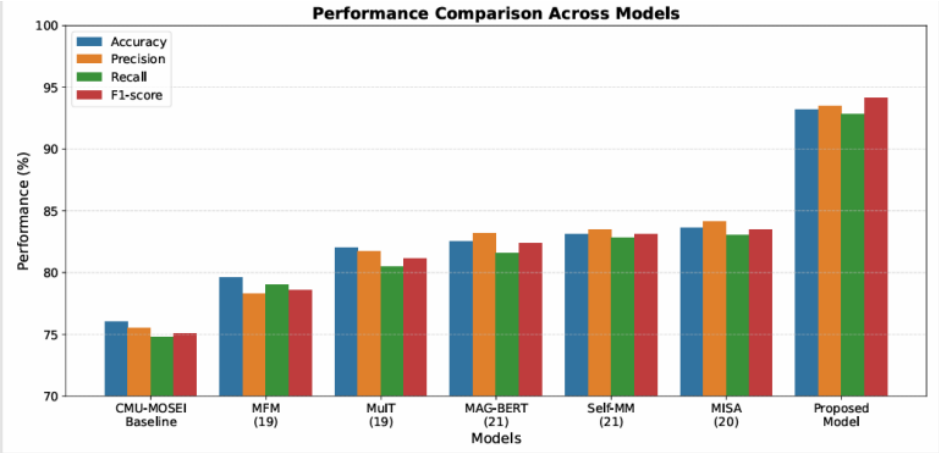


Fig. 9. Performance comparison of classification methods.

F. Discussion

In this study, a new way of performing multimodal sentiment analysis is introduced to deal with challenges such as interpretability and performance. Since RoBERTa is for text, Wav2Vec2.0 for audio, and (ViT) for images, the model is able to detect complex and detailed emotional signals in different types of inputs. The presented method contrasts the previous ones since it employs a hierarchical attention module that considers both within- and between-modal events. So, it is able to pay close attention to features about emotion and modify attention depending on how reliable and relevant each data type is. The automatic adjustment system for different modalities makes the model strong, as providing faulty (like noisy sound or difficult to see the face) inputs will not cause significant problems. Making a model understandable is improved through attention heatmaps, attribution scores, and visual explanations of each patch, so that users can see how the model works. Where healthcare, education, or social media monitoring is involved, being clear about the system's behavior is extremely important. Studying the CMU-MOSEI benchmark confirms that the model surpasses other approaches by attaining a 10% rise in main scores. It is clear from these results that using deep pretrained encoders, adaptive attention fusion, and interpretability makes this approach powerful and trustworthy for the analysis of sentiment in the real world. The proposed work is restricted to CMU-MOSEI and the English language and, therefore, limits cross-domain generalization. The model is demanding of computational resources [33] and thus does not allow deployments in real-time. Although attention heatmaps enhance interpretability, they do not provide a full explanation of human complex reasoning, especially sarcasm or subtle emotions, and thus, it has room for greater future generalizations.

V. CONCLUSION AND FUTURE WORK

This study concludes that adding hierarchical attention mechanisms to the proposed adaptive transformer helps both the accuracy and the explainability of multimodal sentiment analysis. When all these models work together in the framework of multi-level attention fusion, the model proves to have a good ability to find and express emotions in content from various senses. Using transparent decision-making with attention visuals, it is well-suited for demanding situations such as understanding customer opinions, monitoring people's emotional well-being, and improving human-computer interaction, where reasons behind the decisions are important as well as the answers. Going forward, many exciting research areas appear. Improving the model's applicability in many areas could be done by training it with multilingual examples. Besides, including information about a user's past feelings, age, and behavior in social media can make the model more individualized and useful. The deployment of models on many devices will become easier if their versions are made lighter, as

this reduces the number of resources needed. If few-shot learning approaches and domain adaptation are used, it can greatly improve the results in low-data situations. To sum up, this study greatly contributes to the development of strong and understandable multimodal sentiment systems, supporting ethical NLP and setting a solid starting point for flexible applications.

REFERENCES

[1] A. Yadav and D. K. Vishwakarma, "Sentiment analysis using deep learning architectures: a review," Artif. Intell. Rev., vol. 53, no. 6, pp. 4335–4385, 2020.

[2] A. Rahali and M. A. Akhloufi, "End-to-end transformer-based models in textual-based NLP," Ai, vol. 4, no. 1, pp. 54–110, 2023.

[3] S. A. Waheeb, "Multi-Task Aspect-Based Sentiment: A Hybrid Sampling and Stance Detection Approach," Appl. Sci., vol. 14, no. 1, p. 300, 2023.

[4] B. Paneru, B. Thapa, and B. Paneru, "Sentiment Analysis of Movie Reviews: A Flask Application Using CNN with RoBERTa Embeddings," Syst. Soft Comput., p. 200192, 2025.

[5] F. Sufi, "Generative pre-trained transformer (GPT) in research: A systematic review on data augmentation," Information, vol. 15, no. 2, p. 99, 2024.

[6] P. D. Michailidis, "A Comparative Study of Sentiment Classification Models for Greek Reviews," Big Data Cogn. Comput., vol. 8, no. 9, p. 107, 2024.

[7] G. Udahemuka, K. Djouani, and A. M. Kurien, "Multimodal Emotion Recognition using visual, vocal and Physiological Signals: a review," Appl. Sci., vol. 14, no. 17, p. 8071, 2024.

[8] H. Wang, X. Li, Z. Ren, M. Wang, and C. Ma, "Multimodal sentiment analysis representations learning via contrastive learning with condense attention fusion," Sensors, vol. 23, no. 5, p. 2679, 2023.

[9] D. Jayakody et al., "Instruct-DeBERTa: A Hybrid Approach for Aspect-based Sentiment Analysis on Textual Reviews," ArXiv Prepr. ArXiv240813202, 2024.

[10] A. Das, O. Sharif, M. M. Hoque, and I. H. Sarker, "Emotion classification in a resource constrained language using transformer-based approach," ArXiv Prepr. ArXiv210408613, 2021.

[11] J. M. Pérez, D. A. Furman, L. A. Alemany, and F. Luque, "RoBERTuito: a pre-trained language model for social media text in Spanish," ArXiv Prepr. ArXiv211109453, 2021.

[12] C. Petridis, "Text classification: Neural networks vs machine learning models vs pre-trained models," ArXiv Prepr. ArXiv241221022, 2024.

[13] S. Ganguly, S. N. Morapakula, and L. M. P. Coronado, "Quantum natural language processing based sentiment analysis using lambeq toolkit," in 2022 Second International Conference on Power, Control and Computing Technologies (ICPC2T), IEEE, 2022, pp. 1–6.

[14] O. E. Ojo, H. T. Ta, A. Gelbukh, H. Calvo, O. O. Adebanji, and G. Sidorov, "Transformer-based approaches to sentiment detection," in Recent Developments and the New Directions of Research, Foundations, and Applications: Selected Papers of the 8th World Conference on Soft Computing, February 03–05, 2022, Baku, Azerbaijan, Vol. II, Springer, 2023, pp. 101–110.

[15] O. Alagöz and T. Uçkan, "Text Clustering with Pre-Trained Models: BERT, RoBERTa, ALBERT and MPNet," NATURENGS, vol. 5, no. 2, pp. 37–46, 2024.

[16] L. Bacco, A. Cimino, F. Dell'Orletta, and M. Merone, "Explainable sentiment analysis: a hierarchical transformer-based extractive summarization approach," Electronics, vol. 10, no. 18, p. 2195, 2021.

[17] I. Perikos and A. Diamantopoulos, "Explainable Aspect-Based Sentiment Analysis Using Transformer Models," Big Data Cogn. Comput., vol. 8, no. 11, p. 141, 2024.

[18] S. Jaradat, M. Elhenawy, R. Nayak, A. Paz, H. I. Ashqar, and S. Glaser, "Multimodal Data Fusion for Tabular and Textual Data: Zero-Shot, Few-Shot, and Fine-Tuning of Generative Pre-Trained Transformer Models," AI, vol. 6, no. 4, p. 72, 2025.

[19] J. R. Jim, M. A. R. Talukder, P. Malakar, M. M. Kabir, K. Nur, and M. F. Mridha, "Recent advancements and challenges of NLP-based sentiment analysis: A state-of-the-art review," Nat. Lang. Process. J., p. 100059, 2024.

[20] N. J. Prottasha et al., "Transfer learning for sentiment analysis using BERT based supervised fine-tuning," Sensors, vol. 22, no. 11, p. 4157, 2022.

[21] M. Olivato, L. Putelli, N. Arici, A. E. Gerevini, A. Lavelli, and I. Serina, "Language Models for Hierarchical Classification of Radiology Reports with Attention Mechanisms, BERT and GPT-4," IEEE Access, 2024.

[22] N. Alturayeif, H. Luqman, and M. Ahmed, "Enhancing stance detection through sequential weighted multi-task learning," Soc. Netw. Anal. Min., vol. 14, no. 1, p. 7, 2023.

[23] LP_MultiComp_Admin, "CMU-MOSEI Dataset | MultiComp," MultiComp | MultiComp Lab's mission is to build the algorithms and computational foundation to understand the interdependence between human verbal, visual, and vocal behaviors expressed during social communicative interactions. Accessed: Aug. 20, 2025. [Online]. Available: http://multicomp.cs.cmu.edu/resources/cmu-mosei-dataset/

[24] M. Berglund and B. van der Merwe, "Formalizing BPE tokenization," ArXiv Prepr. ArXiv230908715, 2023.

[25] Z. Liu, M. Chen, Z. Wu, Q. Liu, J. Yang, and L. Xie, "Sequence Model with Self-Adaptive Sliding Window for Efficient Spoken Document Segmentation," ArXiv Prepr. ArXiv210709278, 2021, [Online]. Available: https://arxiv.org/abs/2107.09278

[26] J. C. Timoneda and S. V. Vera, "BERT, RoBERTa, or DeBERTa? Comparing Performance Across Transformers Models in Political Science Text," J. Polit., vol. 87, no. 1, pp. 347–364, 2025.

[27] F. T. Lima and V. M. Souza, "A large comparison of normalization methods on time series," Big Data Res., vol. 34, p. 100407, 2023.

[28] G. Ioannides and V. Rallis, "Real-time speech enhancement using spectral subtraction with minimum statistics and spectral floor," ArXiv Prepr. ArXiv230210313, 2023.

[29] Y.-T. Yeh, J. Ock, and A. B. Farimani, "Text to Band Gap: Pre-trained Language Models as Encoders for Semiconductor Band Gap Prediction," ArXiv Prepr. ArXiv250103456, 2025.

[30] P. Sudarsanam, I. Martín-Morató, and T. Virtanen, "Representation learning for semantic alignment of language, audio, and visual modalities," ArXiv Prepr. ArXiv250514562, 2025.

[31] E. Hassan, A. S. Talaat, and M. Elsabagh, "Intelligent text similarity assessment using Roberta with integrated chaotic perturbation optimization techniques," J. Big Data, vol. 12, no. 1, pp. 1–33, 2025.

[32] S. Sathyanarayanan and B. R. Tantri, "Confusion matrix-based performance evaluation metrics," Afr. J. Biomed. Res., vol. 27, no. 4S, pp. 4023–4031, 2024.

[33] L. Qiu et al., "Dynamically Fused Graph Network for Multi-hop Reasoning," in Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, A. Korhonen, D. Traum, and L. Màrquez, Eds., Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 6140–6150. doi: 10.18653/v1/P19-1617.

[34] S. Sun, G. Xu, and S. Lu, "MFM: Multimodal Sentiment Analysis Based on Modal Focusing Model," in 2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC), IEEE, 2024, pp. 1524–1529.

[35] Y. Li, A. Liu, and Y. Lu, "Multi-level language interaction transformer for multimodal sentiment analysis," J. Intell. Inf. Syst., vol. 63, no. 3, pp. 945–964, 2025.

[36] X. Zhao, Y. Chen, W. Li, L. Gao, and B. Tang, "MAG+: An extended multimodal adaptation gate for multimodal sentiment analysis," in ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2022, pp. 4753–4757.

[37] Z. Chen, C. Lu, and Y. Wang, "Self-attention mechanism prior to modality fusion for multimodal sentiment analysis," Multimed. Syst., vol. 31, no. 4, pp. 1–14, 2025.

[38] S. Patel, N. Shroff, and H. Shah, "Multimodal sentiment analysis using deep learning: a review," in International Conference on Advancements in Smart Computing and Information Security, Springer, 2023, pp. 13–29.

A Data-Driven Approach to Achieve Low-Carbon Building Energy Optimization by Using BIM Technology

Xin Yu*, Guoliang Ren, Jie Niu
Changzhou Vocational Institute of Engineering Jiangsu, 213364, China

***Abstract*—Low-carbon building energy optimization is the environmental aspect that is impacted due to the construction sector. The integration of Building Information Modelling (BIM) with Artificial Intelligence (AI) techniques offers the design and operation of buildings with a reduced carbon footprint. Usually, such systems lack flexibility and the precision of dynamically optimizing energy usage. This work proposes a novel data-driven framework that merges AI and BIM to optimize the building energy system for low-emission design using Carbon Major Emission Datasets. It aids materials and energy source selections by identifying highly emitting commodities to reduce operational carbon footprints. Initially, data acquisition and emission analysis are performed on the Carbon Majors database to identify high-emission materials. Subsequently, emission factors are linked with the BIM elements using plug-ins such as One Click LCA, which allow the annotation of embodied carbon values. Further, operational energy is optimized by Multi-Agent Assisted NSGA-II, which optimizes parameters and material selection. Additionally, AI-assisted energy prediction supported by the SqueezeNet model and energy simulation techniques was used for minimizing building energy consumption. The results reveal a high-energy prediction accuracy of 0.0212 for MAE, 0.0376 for MSE, and 0.9814 for the R^2 score. It further helps to reduce carbon emissions by 1155 tons and improve the cost efficiency of 570.25 million, promoting low-carbon building solutions from the earliest stages of design.**

Keywords*—Low-carbon buildings; building information modelling; carbon emissions; operational energy optimization; SqueezeNet; energy simulation

I. Introduction

Low-carbon building in energy optimization is the strategic design, construction, and operation of buildings with the goal of minimizing carbon dioxide (CO_2) and greenhouse gas emissions throughout their life cycle [1]. This method is in reaction to the huge contribution of the built environment toward global climate change, whereby buildings contribute around 39% to global annual CO_2 emissions, 28% from building operations, and 11% from materials and construction (embodied carbon) [2]. The urbanization, industrialization, and energy demand across the world have increased the carbon intensity of building infrastructure over the last four decades [3]. Perceiving climate change mitigation as a global agenda, the Paris and the United Nations sustainable development goals stresses the urgent necessity of decarbonizing the construction sector. More than 75% of the building stock in 2050 will be built today; hence, the optimization of new and existing buildings is paramount [4]. This leads further to the development of advanced energy modelling and carbon accounting tools with policy-inspired benchmarks for the reduction of the lifecycle carbon footprint of buildings [5]. Over the past 15 to 20 years, the emergence of carbon emission trajectories in climate models has brought into sharp relief the prominence of energy optimization for low-carbon building and thereby situated it as one of the key pillars in sustainable urban development worldwide [6].

Due to the increasing demand for sustainable construction, Building Information Modelling (BIM), combined with low-carbon building technology, has been identified as a transformational approach [7]. A BIM system is a digital representation of a building's physical and functional attributes through a 3D model-based process, thus facilitating multidisciplinary collaboration among design, construction, and operation phases [8]. The carbon emission data, along with energy performance, could be embedded into the BIM platform, enabling stakeholders to simulate, analyze, and optimize design alternatives [9]. Price types and potential hybrid modes of construction based on carbon emissions could be informed by such integration [10]. In 15 to 20 years, BIM will graduate to a platform-based 5- and 6-dimensional solution [11]. This has propelled BIM ahead of its contemporaries regarding carbon-aware building methods, hence providing a striking tool for the reduction of embodied and operational carbon footprints early in scheme planning and design, an important activity since over 70% of a building's lifecycle cost and impact are determined during the initial phases [12].

A rapid rise in the practice of BIM-based low-carbon design optimization is being witnessed in developed and emerging countries, owing to varied regulations on carbon, increased concern regarding climate risk, and growth in digital technologies for construction [13]. BIM mandates and sustainability frameworks have been adopted by governments and industries around the globe, with carbon performance increasingly within these frameworks [14]. The incorporation of historical datasets of carbon emissions within BIM environments enables architects and engineers to trace the carbon intensity of a material (such as cement) or goods (such as coal) to the producers for emission-conscious supply chain decisions [15]. This data-oriented integration of BIM and Artificial Intelligence (AI) -based optimizers allows for-going carbon assessments and predictive modelling of energy consumption to be evaluated with surgical precision [16]. This way, it enables buildings to comply with environmental standards, maintain energy efficiency, and sustainable building

*Corresponding Author.

design solutions, thereby drastically cutting carbon emissions and green-resilient structures towards the building agenda. This AI has helped in considerably minimizing carbon emissions, enhancing design efficiency, saving costs through energy optimization, improving regulatory compliance, and creating environmentally resilient and future-ready buildings [17]. Thus, this work provides a data-driven framework to integrate BIM technology into historical carbon emission data through the AI-aided analysis of energy performance and support for low-carbon design decisions.

Research Questions:

1) How can Building Information Modelling (BIM) be integrated with historical carbon emission datasets to support low-carbon material selection in early design stages?

2) In what ways can Artificial Intelligence (AI), particularly deep learning models such as SqueezeNet, improve the accuracy of building energy consumption prediction?

3) How can a multi-objective optimization approach (MAA-NSGA-II) simultaneously balance energy efficiency, carbon reduction, comfort, and cost in building design?

4) To what extent can the proposed AI-BIM framework achieve both embodied and operational carbon optimization across the entire building lifecycle?

The major contributions of the proposed work are:

- Formulate a data-driven framework where the AI-BIM is integrated with the Carbon Majors Emissions Dataset to orient low-carbon building energy optimization about material selection and operational strategies through historical emissions intelligence.
- Analyse carbon emission data to aggregate emissions from coal, cement, oil, and gas to parent entities to identify high-emission materials for sustainable construction planning.
- Integrate emissions data with BIM by connecting emission coefficients with building components with One Click LCA (life cycle assessment) to enable the visualization and annotation of embodiment carbon values within the BIM environments.
- Optimize operational carbon emissions using a Multi-Agent-Assisted NSGA-II (MAA-NSGA-II) algorithm by considering the simultaneous positioning of multiple design parameters and shading configurations for multi-objective energy efficiency.
- Simulate predictive energy demands with an AI-based model that characterizes building energy consumption using a SqueezeNet model for compressing buildings and assessing actual operational energy demand, while maintaining thermal comfort and indoor quality.

The study is organized into sections as follows: Section II is concerned with the study of related literature and existing approaches. Section III describes the AI-based simulation methodology proposed. Section IV provides analysis and discussion of the results in Section V. Finally, Section VI draws the conclusions.

II. Literature Survey

Recently, the convergence of Internet of Things (IoT), AI, and big data analytics has immensely expanded the functional scope of BIM applied in construction and visualization. According to Ho et al. [18], a data-centric approach with ant colony optimization method for low-carbon prefabricated component production is introduced. The study highlighted the importance of prefabrication in reducing environmental impacts in the construction sector. In the same vein, Razi et al. [19] used a multi-objective prediction model for time, cost, energy consumption, and CO_2 emissions. Applying decision-making methods to identify influencing factors and utilizing eleven machine learning (ML) algorithms, the study forecasted the four competing priorities and balanced them. Another work by Popa et al. [20] developed a predictive tool that cohered energy consumption data under the IoT to assign energy performance ratings. Their model tested the capability of an efficiency label for buildings by using minimal input features of floor area. Between these developments, Myintet al. [21] focused on carbon estimation during numerous stages of construction, including raw material processing and transportation, by using data acquired from a residential project. Yet, challenges in integrating emission data and allow for a holistic low-carbon optimization in BIM workflows.

BIM has been transformative in embedding energy performance and carbon footprint assessment throughout a building's lifecycle. For instance, Zhao et al. [22] developed a BIM approach with 3D laser scanning to generate energy models for the building. The method was employed to analyze retrofit options to convert existing buildings into operational nearly zero-energy systems. Similarly, Shen et al. [23] introduced a full-life-cycle net-zero carbon building framework. Their work utilized an ontology-based approach in BIM to delineate key decision variables that cover design, construction, and operation, thus allowing seamless data integration and decision-making based upon it. Also, Tahmasebinia et al. [24] utilized regression models within BIM environments to simulate energy-efficient design. The study showed that multi-linear regression was used to estimate energy consumption for different architectural shapes and materials, revealing that triangular shapes produced the best energy performance. Another survey by Zhuang et al. [25] explored a performance-oriented BIM framework to optimize environmental impact and energy usage throughout the building life cycle. They employed a school building to analyze the effects of different envelope structures on indoor environment conditions and cost-effectiveness. However, lacked dynamic integration of emissions data to enable true holistic optimization.

Advancements in data-driven technologies, especially AI, BIM, and big data analytics, have increased opportunities for optimizing the energy performance of buildings. Mehraban et al. [26] studied residential buildings' energy behaviour in hot climatic zones by applying a BIM-ML approach. The approach involved simulating energy use via platforms such as green building studio and insight with respect to building orientation, fenestration, floor area, wall constitution, infiltration rates, and daylighting. Four ML methodologies were used to predict energy use intensity and thus generated information on the design's energy use. Likewise, Wang et al. [27], Arsiwala [28],

and Giannelos et al. [29], among others, have studied carbon emission prediction at the initial stages of construction, focusing on integrating BIM-ML techniques. They studied 35 public buildings in China to measure emissions caused by material selection. In parallel, digital twin models combined with ML were employed to engineer the monitoring of CO_2 emissions from operational buildings. Also, they used ML to predict emission levels at various stages of the life cycle in the building sector. Despite significant achievements, many of the solutions faced limitations with respect to predicting emissions across varied climates, materials, and emission scenarios, hence paving the way for generalizable emissions-informed BIM-AI frameworks. Table I presents the limitations of previous studies and the solutions proposed in this study.

TABLE I. LIMITATIONS OF PREVIOUS STUDIES AND SOLUTIONS PROPOSED IN THIS STUDY

Study	Limitations	Proposed Solutions
Ho et al. [18]	Looked only at parts, not the full building carbon.	Covers both building materials and energy use with AI.
Razi et al. [19]	Balanced cost, time, energy, but not a full low-carbon design.	Optimizes energy, emissions, and comfort together.
Popa et al. [20]	Used very few inputs; no detailed carbon data for materials.	Adds detailed carbon data of materials into BIM for better design.
Myint et al. [21]	Did not fully connect carbon data in BIM.	Links carbon data directly in BIM with One Click LCA.
Zhao et al. [22]	Focused only on retrofit energy, not the full life cycle.	Handles both embodied and operational carbon for the full life cycle.
Shen et al. [23]	Did not use changing data during building use.	Uses AI for dynamic updates over the life of the building.
Tahmasebinia et al. [24]	Looked at energy, but not much carbon.	Combines both energy and carbon.
Zhuang et al. [25]	No real-time adjustment for energy and carbon.	Adds AI for real-time changes across the life cycle.
Mehraban et al. [26]	Only studied hot-climate houses.	Uses a global dataset, so it works in all regions.
Wang et al. [27]	Looked only at carbon at construction start.	Covers both short-term and long-term carbon with AI + BIM.

A. Problem Statement

Despite advances in integrating BIM with data-driven technologies, it still has not been able to develop fully optimized low-carbon building energy systems. Most of the methods consider operational carbon and embodied carbon, but they are not linked with emission data from materials such as cement, coal, and natural gas, known to be major contributors to the carbon footprint of the building sector [30]. Also, they present different explanations of energy simulation tools that exert generalized assumptions and do not provide internalized insight into carbon-intensive commodities, compromising their early-stage design decisions [31]. Retrofit models are also limited by small datasets, lacking dynamic adjustment through building lifecycles, or are limited to a narrow geography and climatic condition [32]. Besides, AI methodologies are used for energy prediction but rarely consider carbon emission data, especially from databases based on emissions of global actors [33]. Therefore, a clear gap exists as existing research either remains limited to operational optimization, neglect material-level carbon data, or fail to integrate dynamic lifecycle adjustments. Few approaches combine embodied and operational carbon within a unified BIM-AI framework, which restricts their real-world applicability. To overcome these challenges, this study proposes a novel data-driven methodology that integrates BIM and AI, with an infusion of historical emissions data to define low-carbon decisions throughout the building life cycle, achieving both embodied and operational carbon optimization.

III. PROPOSED AI-BIM LOW-CARBON ENERGY FRAMEWORK

This framework presents a data-driven low-carbon building energy optimization framework integrating BIM with historical emissions data and AI-trained optimization techniques. The proposed algorithms are especially effective when applied to detailed and high-dimensional datasets, such as BIM models enriched with material-level carbon factors and time-series operational data (e.g., HVAC loads, occupancy, lighting). Their strength lies in handling multi-variable scenarios where both embodied and operational carbon must be optimized together, whereas performance may be less comprehensive when applied to small or low-detail datasets. Utilizing the carbon emissions dataset allows identify the commodities with high emissions and trace the carbon responsibility to particular entities. The framework starts with an investigation of emissions, wherein data is aggregated by commodity and entity to identify high embodied carbon materials. This is followed by a data embedding process into BIM environments, which associates carbon coefficients with building components and thus allows for graphical exposition of environmental impact. Similarly, an MAA-NSGA-II algorithm is used to perform multi-objective optimization of crucial operational parameters, while the AI-assisted prediction model simulates building energy use to reduce operational emissions and work alongside human comfort to design carbon-aware energy-efficient buildings. The general synthesis of the proposed framework is shown in Fig. 1.

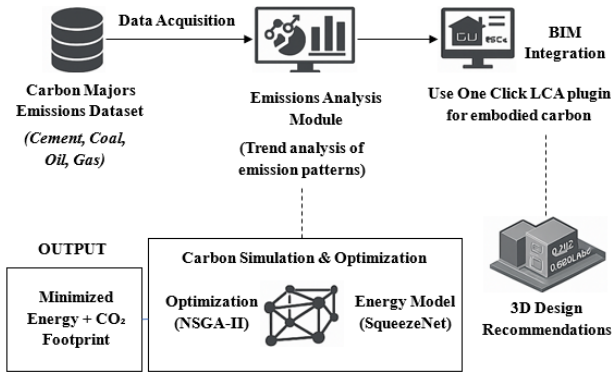


Fig. 1. AI-BIM Integrated framework for low-carbon building energy optimization.

A. Data Acquisition

The dataset adopted in this work consists of carbon emissions, providing a history of emissions data from 1854. The data are provided by the Climate Accountability Institute. It compiles production and emissions data from 122 major fossil fuel and cement producers. This database was selected due to its being one of the most authoritative and comprehensive in

existence, reporting over 72% of global fossil fuel and cement emissions since the onset of the industrial era. Its broad coverage and granularity by commodity and producer make it incredibly well-suited to associating emission factors with BIM components and facilitating sound low-carbon design choices. The data for emissions analysis were obtained from corporate disclosure, national inventory, and academic research, guaranteeing reliability. This study entails the variables year, parent entity, commodity type (coal, cement), production quantity, and total emission observed in $MtCO_2e$ to capture emission-intensive materials and producers over time.

B. Emissions Analysis

During this stage, the emissions data is accumulated and analyzed to build a carbon-intensive materials and emission contributors. Emissions are grouped by type of commodity, specifically by coal, cement, oil, and gas, on parent entity ownership models considered by type of ownership from the investor. This sets the basis for understanding with level of granularity with which materials and producers have contributed in the past toward global carbon emissions. In the analysis, it produced material-wise emission coefficients towards informing low-carbon design in BIM environments. These coefficients serve as quantified indicators of embodied carbon, allowing architects and engineers to identify and consequently circumvent or lessen high-emission materials at the conceptual design phase level.

This study, using the aggregation of measurements provided by the dataset (1854- Present), identifies carbon-intensive materials by a detailed dataset from commodity type and parent entity (state-owned versus investor-owned). These coefficients guide the annotation of BIM components toward carbon-conscious design and planning (for instance, annotations may encourage the use of Portland cement substitutes consisting of lower embodied carbon).

C. BIM Integration

Once the emissions analysis is conducted, the framework incorporates emission intelligence into the BIM environment. This is done through One Click LCA, the premier platform for LCA and embodied carbon calculations, used to make environmental data, mainly emission coefficients for high-carbon materials like cement, coal-based steel, and fossil fuel derivative chemicals, feed into design decisions on the digital stage. Incorporation of emissions data in the LCA platform follows a multi-step procedure,

1) Initially, emission coefficients are retrieved from the dataset. The coefficients stand for average carbon emissions per unit of material. For instance, 0.95 $MtCO_2e$ per tonne of cement. The materials are arranged according to their commodity type, production source, and historical emissions; thus, they represent more realistic environmental impact metrics.

2) A highly detailed digital building model was imported from Autodesk Revit into One Click LCA via a plugin. The plugin reads and interprets the BIM model in terms of basic structural and architectural elements, and it extracts quantities and dimensions, material specifications, and location-based metadata for all components (such as slabs, beams, insulation, and facades).

3) One Click LCA features an extensive environmental database, carrying thousands of building materials around the world and their environmental profiles. It automatically maps BIM materials to corresponding emission profiles so that tailored emission coefficients can be manually assigned to each material element in the model.

4) After materials get mapped, One Click LCA calculates the embodied carbon for each building component by multiplying the emission factor by the quantity extracted from the BIM model. Subsequently, these values are annotated into the respective model elements, thus enabling the user to visualize the particular components, contributing to the overall carbon footprint of the building.

5) Finally, this tool also prepares detailed LCA reports and carbon impact dashboards showing total embodied emissions of the building, material or construction-phase-wise breakdowns, and comparison across design alternatives. Such information is then available to architects and sustainability engineers to make informed decisions, based on data, either on low-carbon cement alternatives or structural systems optimization from a carbon perspective.

By embedding emission coefficients and embodied carbon within the BIM environment, this integration elevates the static 3D models into living, carbon-conscious digital twins. The initiative enables stakeholders to simulate and optimize the environmental performance of a building before embarking on physical construction, therefore supporting the overarching goal of a design that leads to energy-efficient, low-carbon buildings from the emissions data.

D. Carbon Simulation and Optimization

Further improvements in building energy efficiency and operational carbon emission reduction are achieved by considering the use of MAA-NSGA-II within the proposed framework. This multi-objective optimization within a BIM environment addresses critical design and operational parameters, enabling maximizing conflicting objectives at the same time, such as minimizing energy consumption and maximizing occupant comfort and material sustainability. The overall system of the MAA-NSGA-II optimization outline is shown in Fig. 2.

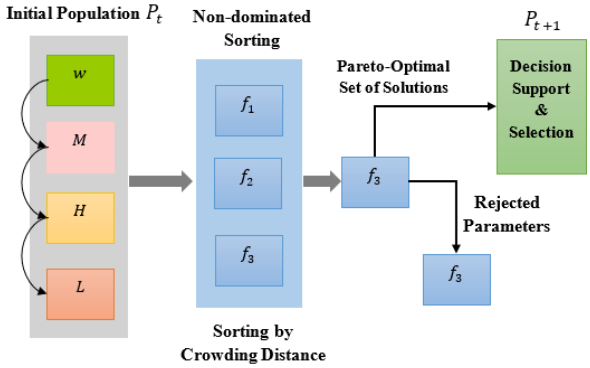


Fig. 2. Illustration of the MAA-NSGA-II optimization framework.

The primary parameters selected for optimization include:

- Window size and orientation (affecting solar gain and daylight utilization).
- Building envelope materials and insulation levels.
- HVAC system configuration and efficiency.
- Lighting and shading strategies.

The initial population of candidate design vectors ($\vec{X}$) for the design alternatives is mathematically modelled as in Eq. (1):

$$\vec{X} = [x_1, x_2, x_3, x_4] = [W, M, H, L] \quad (1)$$

where, $x_1 = W$ signifies window sizes and orientations, $x_2 = M$ includes building materials and their insulation parameters, $x_3 = H$ represents types and configurations of HVAC systems, and $x_4 = L$ represents lighting strategies and shading devices. These variables influence energy demand and emissions through the building life cycle. This optimization is conducted to minimize the operational energy and consequently emissions of the building while maximizing thermal comfort. Then, the mathematical formulation of this multi-objective optimization is indicated in Eq. (2):

$$\min_{\vec{X}} \begin{cases} f_1(\vec{X}) = E_{\text{total}}(\vec{X}) \\ f_2(\vec{X}) = C_{op}(\vec{X}) \\ f_3(\vec{X}) = -C_{\text{int}}(\vec{X}) \end{cases} \quad (2)$$

Here, $f_1(\vec{X})$ quantifies total energy consumption (E_{total}) in kWh, $f_2(\vec{X})$ reflects carbon emissions (C_{op}) in kilograms of CO_2-equivalent, and $f_3(\vec{X})$ captures the inverse of the comfort function (C_{int}), where represented using the predicted mean vote to ensure indoor thermal satisfaction.

Energy use throughout the period of operation of the building is dynamically simulated by Eq. (3):

$$E_{\text{total}}(\vec{X}) = \int_0^T \left(Q_{HVAC}(t) + Q_{\text{lighting}}(t) + Q_{\text{equipment}}(t)\right)dt \quad (3)$$

where, $Q_{\text{HVAC}}(t)$, $Q_{\text{lighting}}(t)$, and $Q_{\text{equipment}}(t)$ represent time-variant energy loads of respective systems over a total simulation time T. These load profiles are compressed and represented using an AI model, enabling efficient processing within multi-agent systems. Then the operational carbon emissions are directly related to the energy consumed by various systems, weighted by their emission factors. This relationship is formalized as in Eq. (4):

$$C_{op}(\vec{X}) = \sum_{i=1}^{n} (\eta_i \cdot E_i) \quad (4)$$

where, E_i is the energy usage of the i^{th} building subsystem (lighting), and η_i is the corresponding CO_2e emission factor for grid-supplied electricity (n). This provides a direct estimation of emissions based on design choices.

To ensure occupant well-being, the thermal comfort function $C_{\text{int}}(\vec{X})$ is evaluated in the model embedded within a thermal comfort agent, as shown in Eq. (5):

$$C_{int}(\vec{X}) = f(T_a, T_r, RH, V_a, G, I_{cl}) \quad (5)$$

where, T_a and T_r denote air and radiant temperatures, respectively, RH is relative humidity, V_a is air velocity, G is the metabolic rate, and I_{cl} is the clothing insulation index. The function $f(\cdot)$ computes the score to estimate comfort from -3 (cold) to +3 (hot). Correspondingly, multiple intelligent agents (m), each with their own utility functions $\phi_j(\vec{X})$, evaluate candidate designs in parallel. These evaluations are then aggregated to compute a global fitness score using a weighted summation (Φ), as expressed in Eq. (6):

$$\Phi(\vec{X}) = \sum_{j=1}^{m} \omega_j \cdot \phi_j(\vec{X}) \quad (6)$$

Here, ω_j are user-defined weights reflecting the priority of each agent (comfort, cost, emissions), and $\phi_j(\vec{X})$ are individual agent outputs. This enables modular and adaptable evaluation supporting BIM feedback. The evolutionary facet of MAA-NSGA-II is used to evolve a Pareto-optimal population in time, preserving diversity and convergence. Therefore, the output of the whole process is a solution set of the non-dominated solutions, as defined in Eq. (7):

$$F = (P_t) = \{\vec{X}_1, \vec{X}_2, \ldots, \vec{X}_k\} \in \text{ Pareto Front} \quad (7)$$

where, P_t is the population at generation t, and F is the final set of trade-off solutions between emissions, energy, and comfort. This Pareto front aids architects and engineers in selecting optimal design configurations early in the design process. Algorithm 1 shows the MAA-NSGA-II for low-carbon building optimization.

Algorithm 1: MAA-NSGA-II for Low-Carbon Building Optimization

Input: Initial population ($\vec{X}$)
Output: Final Pareto-optimal front of low-carbon design solutions (F)
Begin
Initialize $\vec{X}$ with random design vectors
For each $\vec{X}$ in Population do
Evaluate $f_1(\vec{X}), f_2(\vec{X}), f_3(\vec{X})$
For each agent m
Compute agent utility: $\phi_j(\vec{X})$
End For
Compute global fitness: $\Phi(\vec{X_i}) \leftarrow \sum_j=1^m (\omega_j \times \varphi_j(\vec{X_i}))$
End For
For $t = 1$ to G do
Perform Non-Dominated Sorting on P_t
Calculate Crowding Distance
Select Parent Set using Tournament Selection
Generate Offspring ϕ_j using Crossover and Mutation
Evaluate objectives and agent utilities for ϕ_j
Select Next Generation P_{t+1}
End For
Extract final Pareto Front (F)
End

E. AI-Assisted Energy Prediction

After optimization of parameters, refined building design parameters, dynamic energy consumption predictions are made

through an AI-assisted approach using the SqueezeNet model, ensuring the best possible accuracy in predicting operational energy usage, which is the direct cause of carbon emissions in the building environment. The overall architecture of the proposed SqueezeNet framework is shown in Fig. 3.

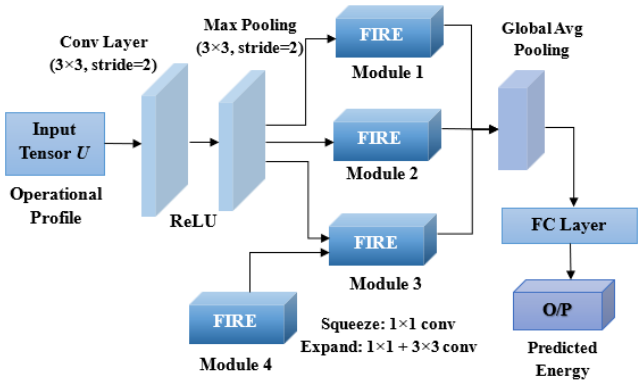


Fig. 3. Architecture of the SqueezeNet for building energy prediction.

The high-dimensional spatiotemporal datasets were considered for training of this model, wherein each input sample represents some operational state of a building over time. The data is in three-dimensional tensor form, $U \in \mathbb{R}^{T \times Z \times F}$, where T is the number of time steps, and Z corresponds to spatial zones inside the building, such as rooms or floors. Also, F represents feature channels detailing contextual variables, including temperature, occupancy levels, lighting usage, HVAC operations, and daylighting contribution. The first convolutional layer in the model extracts low-level spatial features F_0 according to Eq. (8):

$$F_0 = \sigma(W_0 * U + b_0) \tag{8}$$

Here, W_0 learned weights of the convolutional filters, b_0 is the bias term, $*$ is the convolution operator, and σ is the ReLU operator used as an activation function to add non-linearity. These extracted features are the basis on which the subsequent layers of the model will extract more complicated building dynamics.

One of the main aids of the model is the Fire modules, which cut down model complexity to a huge extent while maintaining its performance. Each module includes a squeeze layer with 1×1 convolutions that reduce the number of input channels, followed by an expand layer that applies both 1×1 and 3×3 convolutions. Thus, the Fire module F_i can be computed as follows in Eq. (9):

$$F_i = \phi\left(\text{Conv}_{1\times1}(F_{i-1}) \| \text{Conv}_{3\times3}(F_{i-1})\right) \tag{9}$$

Here, $\|$ represents concatenation, and ϕ is a function that merges the expanded feature maps from both convolution paths. This modular design enables SqueezeNet to drastically reduce parameter count and computation time when integrated within large-scale BIM systems for energy prediction.

After passing through a stack of Fire modules, the final compressed feature representation F_n is forwarded to a regression head with a fully connected output layer that predicts the total energy consumption $\grave{E}_{\text{total}}$. The regression function is formulated, as in Eq. (10):

$$\grave{E}_{\text{total}} = w_e^T \cdot F_n + b_e \tag{10}$$

where, w_e is the learned weight vector for regression, and b_e is the corresponding bias. This ensures that the model iteratively learns to minimize prediction error across diverse operating scenarios. Algorithm 2 presents the AI-assisted energy prediction using SqueezeNet.

Algorithm 2: AI-Assisted Energy Prediction Using SqueezeNet

Input: Optimized Extract parameter vector (F)
Output: Predicted total energy consumption (in kWh)
Begin
Initialize model
Load AI architecture with Fire Modules
For each timestep t from 1 to T do:
Extract U ← usage profile at time T
Apply initial convolution U
For each Fire Module i:
Perform squeeze operation → reduces input channels ($Conv1x1$)
Perform expand operation → parallel $Conv1x1$ and $Conv3x3$
Concatenate results to produce F_i
End For
Aggregate final feature vector F_n from last Fire Module
Compute predicted energy $\grave{E}_{\text{total}}$
Return $\grave{E}_{\text{total}}$
End For
End

This phase is crucial in the optimization of building energy systems and low-carbon architectural design through advanced simulation aided by AI. The system provides an energy usage predictor employing a forecast with the SqueezeNet model. Within it resides the power to make a prior judgment on the configurations that stand to be the least energy-consuming and environmentally sustainable design options. This will make data-driven decisions in the early stages of the design process and take the biggest potential for impact-control on CO_2 emissions away before construction begins.

This intertwining of predictive modelling with BIM enables architects to create incremental design alternatives in a way that operational energy consumption, embodied carbon, indoor thermal comfort, as well as compliance with local standards are considered. It allows designers and engineers to further reduce operational carbon and energy while still maintaining occupant comfort, which is essential in designing and constructing low-carbon and energy-efficient buildings.

IV. Results and Discussions

This study contains the assessment and comparative analysis of the proposed framework for optimizing low-carbon building energy, developed and simulated using Python. Simulation parameters considered building geometry, material emission coefficient, HVAC configuration, insulation level, and solar orientation. Multi-objective optimization was used to balance energy efficiency and carbon reduction, while AI served as a

model to compress usage profiles for energy prediction. The framework identified the design configurations to lower total CO_2 emissions, proving its application in guiding the early design phase for carbon-conscious decisions.

A. Dataset Description

Carbon Majors Emissions Dataset, a globally recognized and authoritative source of historical carbon emission data. Emissions are broadly covered by the dataset from 1854 to date. Its peculiar dataset traces well beyond over 1.42 trillion tons of carbon dioxide equivalent (CO_2e) emissions, almost 72% of global fossil fuel and cement-related emissions since the dawn of the Industrial Revolution in 1751, hence ranking it as one of the most thorough repositories for emissions tracking and attribution.

This dataset is developed for multidimensional analysis while including major variables such as.

- Year of emission record,
- Parent entity and its type (investor-owned, state-owned, or nation-state),
- Commodity type (including Oil, Gas, various Coal types, and Cement),
- Production quantity and units (million barrels, billion cubic feet, or million tons),
- Total emissions (in $MtCO_2e$).

The dataset is curated from producers ranked among the top 122 in the globe in oil, gas, coal, and cement, such as:

- 75 investor-owned companies,
- 36 state-owned companies, and
- 11 nation-states.

It comprises 82 oil producers, 81 gas producers, 49 coal producers, and 6 major cement producers. Apart from this, the dataset is also provided at a low, medium, and high level of granularity, which facilitates scaling analysis from aggregated trends to much finer emissions at the entity and commodity levels. It also facilitated carbon-aware decision-making via dynamic annotation and optimization of building components. Therefore, the dataset acts as a historical record and, concurrently, as an enabler of opposite-looking low-carbon design strategies in a data-driven building energy optimization framework.

Dataset link: Carbon Majors Emissions Data.

B. Performance of Emissions Analysis

This section undertakes a thorough analysis of carbon major emissions data for high-emission materials and responsible entities. This analysis, by aggregating emissions from commodity types and ownerships of the entities, brings to light emission hotspots for low-carbon considerations in BIM design.

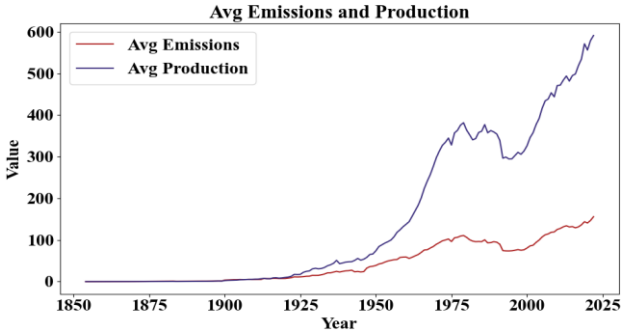


Fig. 4. Time trends of Emissions vs. Production (1854–2024).

Fig. 4 presents the historical average carbon emissions and production volume trends over time, from the Carbon Emissions dataset. This shows steep growth, especially post-1950, with production and emissions soaring with industrialization of fossil fuels. By 2024, average production had peaked at around 584 units, while emissions reached nearly 140 $MtCO_2e$. This peak of carbon emissions and commodities helped these BIM-integrated decisions to lessen carbon impact through informed decisions for the choice of material and source of energy.

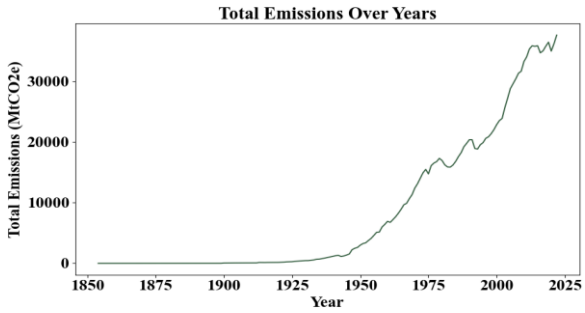


Fig. 5. Historical surge in carbon emissions accumulation.

Fig. 5 exhibits the long-term time trend of total emissions (in MtCOe) from 1850 to 2024. This indicates how emissions have a historical upward trend, becoming drastically steeper after 1950. By 1975, total emissions surpassed 15,000, after which total emissions show even a sharper rise to finally breach the 30,000 $MtCO_2e$ mark by 2020. This corresponds to industrial time, and, thus, brings to attention the need for limiting emissions in the present design and policy framework. Both of these historical viewpoints are aware of the legacy emission loads and planning for data-driven pathways for building environment and energy systems.

Fig. 6 displays the fine-grained analysis of different features contributing to carbon emissions. Product emissions $MtCO_2$ received the highest score of 0.200, followed by commodity type (~0.155), year (~0.125), and production value (~0.100). The fugitive methane emissions $MtCO_2e$ (~0.090), total operational emissions $MtCO_2e$ (~0.080), and parent entity (~0.080). Then there were lesser influential to be noted as own fuel use emissions $MtCO_2$ (~0.065), total emissions $MtCO_2e$ (~0.060), and flaring emissions $MtCO_2$ (~0.050). It helps to elucidate where emphasis is placed in the emission-aware design decisions within the BIM frameworks, thus ensuring that the targeted mitigation measures are applied to high-impact contributors.

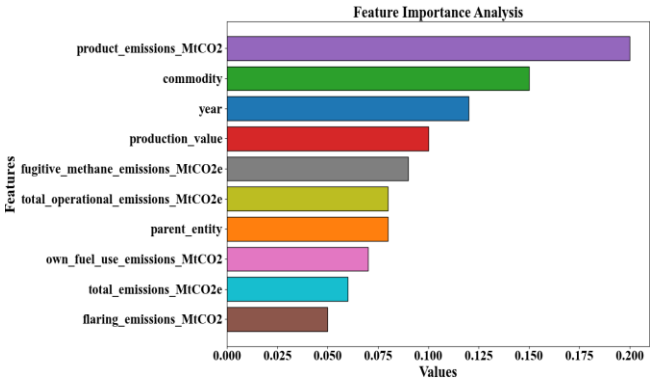


Fig. 6. Feature importance insight of influential drivers in carbon emissions.

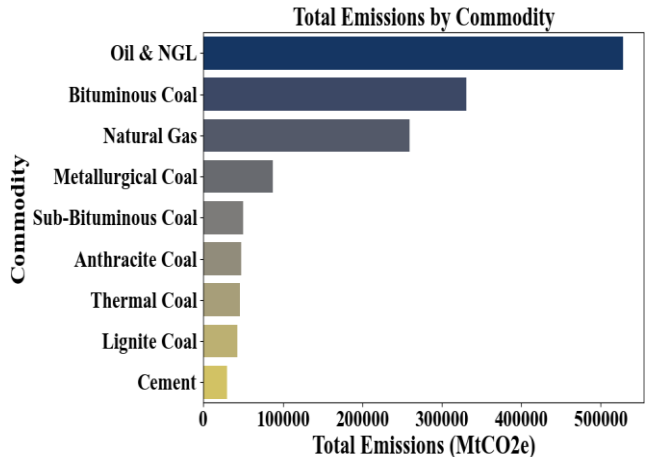


Fig. 7. Emission intensity by commodity – sectoral breakdown of carbon burden.

Fig. 7 provides an analysis of total carbon emissions (in MtCO2e) by commodity type from the dataset. It reveals that Oil & NGL is the most carbon-intensive sector, with emissions of over 500,000, followed by Bituminous Coal at roughly 350,000 and Natural Gas at around 270,000. The carbon emissions include Metallurgical Coal (~130,000), Sub-Bituminous Coal (~75,000), Anthracite Coal, Thermal Coal, and Lignite Coal, each emitting between 50,000. Finally, cement comes under 40,000 MtCO2e. This helps in identifying the most carbon-intensive sectors on which to focus the decarbonization interventions in commodity-specific policies and sustainable design practices.

The embodied carbon emissions for construction materials are summarized in Table II, which shows the emission coefficient and corresponding embodied carbon values for cement, steel, and glass, with steel recording the highest carbon emissions among these materials. Table III shows total carbon emissions globally by parent entity type, thus stressing the implications of material choice in BIM systems, and the effects of organizational ownership on the assignment of global emissions accountability; therefore, this bears on material choice and emission governance strategies in sustainable infrastructure development.

TABLE II. CARBON QUANTIFICATION FROM CONSTRUCTION MATERIALS

Material	Emission Coefficient	Quantity	Embodied Carbon ($MtCO_2e$)
Cement	0.95	1000	950.0
Steel	2.10	500	1050.0
Glass	1.40	200	280.0

TABLE III. TOTAL EMISSIONS BY PARENT TYPE

Parent Type	Percentage of Total Emissions
Nation State	36.3%
State-owned Entity	32.7%
Investor-owned Company	31.0%

C. Performance Evolution

Here, evaluate the aggregate performance of the SqueezeNet integrated optimization framework in minimizing carbon emissions and energy use. It measures improvements through simulation and establishes that data-driven design choices greatly improve the environmental performance of a building initial stages of planning.

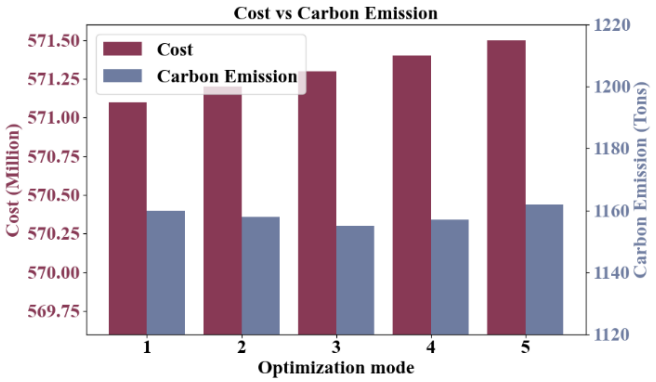


Fig. 8. Multi-objective trade-off between cost and carbon emission across optimization modes.

Fig. 8 analyzes cost (in millions) and carbon emission (in tons) across different optimization modes, based on the proposed BIM-integrated framework using an MAA-NSGA-II approach. Mode 3, thus optimized, recorded the lowest carbon emissions (~1155 tons) at a fairly low cost (~570.25 million), setting an almost-best trade-off. Equally, Mode 5 yields the most expensive costs (~571.45 million), albeit emissions were slightly reduced (~1165 tons), allowing engineers to select the most sustainable and cost-efficient configurations available for early-stage building design.

Fig. 9 compares the actual energy consumption data against predictions from the AI model. With almost exact values, the model proved its ability to accurately interpret patterns of energy consumption from building design parameters and profiles, thus validating its use in early-stage building design for the estimation of operational energy. This predictive capability favors the implementation of early-phase optimizations of energy systems toward a low-carbon design strategy.

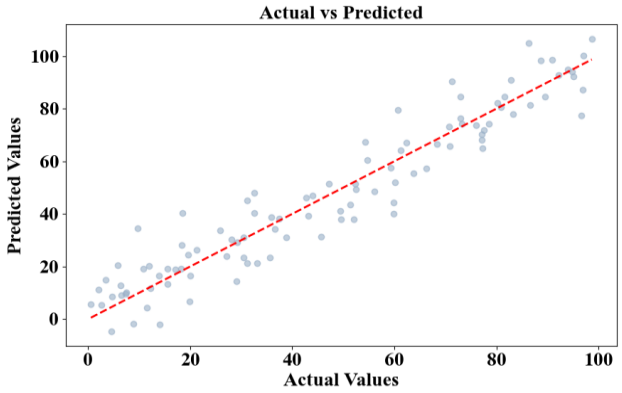


Fig. 9. Energy usage prediction accuracy using SqueezeNet.

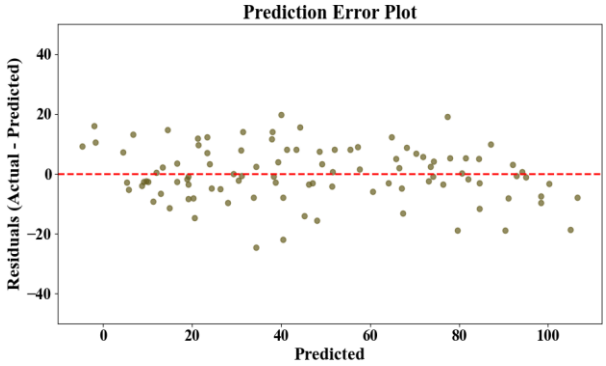


Fig. 10. Prediction error plot of energy consumption.

Fig. 10 shows the distribution of energy consumption prediction errors, highlighting the actual and predicted energy use for the various building configurations. The low clustered error values point to high model precision and minimal error in energy load prediction and thus reflect the model's capability of capturing complicated energy behavior patterns. Such a low error in prediction instils confidence in the designed framework, with attention to energy-efficient considerations in the building design process while being carbon-conscious.

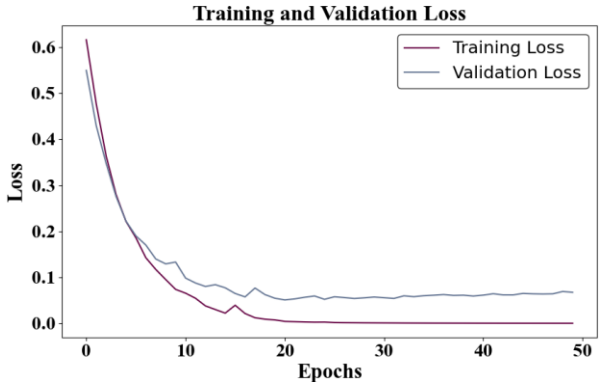


Fig. 11. Training and validation loss curve of the energy consumption.

Fig. 11 represents the training and validation-loss development of the DL model for building energy consumption prediction. The training loss gauges the ability of the model to fit the training data and turns out to be 0.00354. The validation loss, however, is slightly higher at 0.09634, yet it indicates that the predictive capabilities are good and no extensive overfitting. Through monitoring the changes in losses during the model training, it attains a trade-off between accuracy providing reliable energy consumption forecasting for building energy optimization tasks.

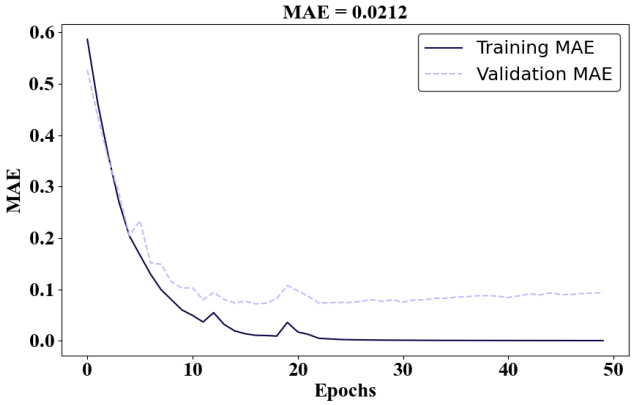


Fig. 12. Training and validation of MAE evolution.

Fig. 12 shows the training and validation Mean Absolute Error (MAE) of the model in predicting building energy consumption. This MAE considers the average magnitude of errors in prediction and ignores their direction. An overall MAE of 0.0212 shows that the model predicted energy consumption values to the actual energy consumption with high reliability.

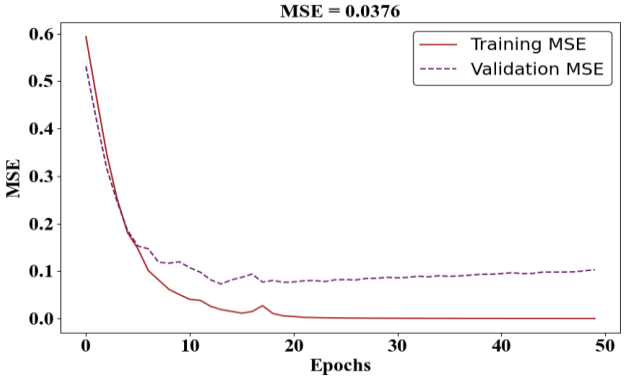


Fig. 13. Training and validation MSE analysis.

Fig. 13 shows the evolution of training and validation of Mean Squared Error (MSE) during the model training process. It quantifies the average squared difference between predicted and actual values, thus giving an indication of model accuracy. The research records an overall final MSE of 0.0376, showing that this predictive model for building energy consumption performs well in predicting the energy use of low-carbon building design.

In Fig. 14, the coefficient of determination (R^2 score) of the predictive model gives the proportion of variance of the observed data explained by the AI model. Along with this, the RMSE (Root Mean Squared Error) measure how far the observation values are from the predicted ones. Thus, the model achieved an overall R^2 score of 0.9814 and the RMSE of 0.1060, showing excellent predictive accuracy that confirms the reliability of the data-driven framework to produce accurate energy consumption.

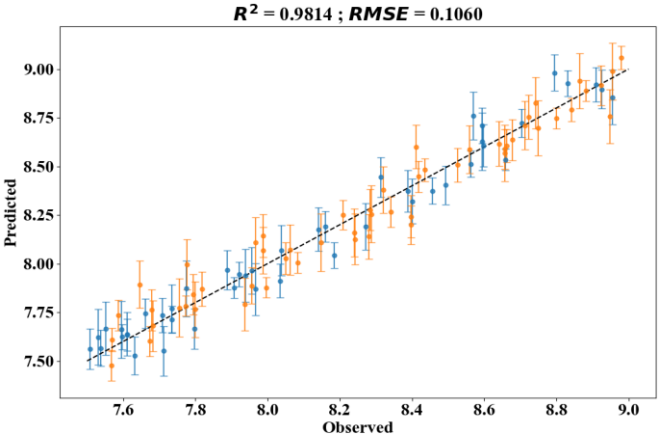


Fig. 14. Model performance evaluation using R^2 score and RMSE.

The above outcomes present the comprehensive analysis of the performance of the proposed framework. For the sake of further interpretation of such results in light of relevant studies and practical usage, the following section includes the discussion.

V. Discussion

A. Comparative Analysis

In this study, the comparative evolution reveals consistent improvements with regard to energy prediction accuracy and carbon reduction, confirming the efficacy of the data-driven AI optimization process through the development stages.

TABLE IV. Comparative Evaluation of AI Models for Carbon Emission Prediction

Models	MAE	MSE	RMSE	R^2 score
GAN [34]	0.2600	-	-	86.00
LSTM [35]	46.10	4570.14	67.60	96.00
GB [36]	24.06	-	35.18	91.00
SqueezeNet	0.0212	0.0376	0.1060	98.14

Table IV provides a comparative analysis between the proposed AI-driven method and state-of-the-art techniques, such as GANs (Generative Adversarial Networks), LSTMs (Long Short-Term Memory), and GBs (Gradient Boosting), for the prediction of carbon emissions from energy consumption forecasts. Compared to these models, the AI-based framework proposed model outperforms them tremendously. In particular, the SqueezeNet-based prediction had the lowest MAE, MSE, and RMSE values, along with the highest R^2 score, proving superior accuracy. Moreover, the integration with MAA-NSGA-II provides multi-objective optimization that balances cost, energy use, emissions, and comfort at the same time, which most existing models fail to address. This accommodation of the model to low-carbon building design through emission forecasting, to the efficient integration in SqueezeNet to account for complex temporal dependencies.

VI. Conclusion

This study effectively developed and implemented an intelligent and data-driven framework that integrates BIM with the SqueezeNet architecture for exact prediction and optimization of building energy consumption. This framework further supports low-carbon construction strategies by allowing the early-stage evaluation of design alternatives, based on the carbon major's emissions dataset. The model was enhanced by incorporating MAA-NSGA-II to balance two conflicting objectives, such as maximization of energy efficiency and minimization of emissions. This analysis revealed that an MAE of 0.0212, an MSE of 0.0376, and an R^2 of 0.9814 were achieved, indicating an impressive accuracy in prediction and generalization capacity. These results are even better than the conventional models, further supporting the energy performance superiority of the proposed method. The framework showcased its practical effect by reducing carbon emissions by 1155 tons while achieving cost savings of about 570.25 million, confirming that it offers both environmental and economic advantages. The comparative analysis also highlighted its superiority over GAN, LSTM, and GB models, establishing the framework as both highly accurate and computationally efficient. This integrated framework comprises reduced computational complexity due to AI-based design considerations, emission feedback aimed at BIM integration, and robust Multi-objective optimization for sustainable architectural decision-making. This work ultimately presents the novel, scalable, and interpretable method inherently applicable to different building typologies, thus assisting in the world shift towards low-carbon city development and climate action decisiveness. Yet, this study does have some limitations since it was primarily validated against historical emission datasets and simulations. How it performs on operational real-time data or under extremely diverse regional contexts still needs to be tested. Future work will improve the framework by using more detailed data and real-time sensors, adding better comfort models for different users and climates, improving occupant behavior prediction with AI, and applying the approach to groups of buildings or whole districts for sustainable city planning.

Funding

Authors do not have any funding.

Conflicts of Interests

Authors do not have any conflicts.

Data Availability Statement

The data generated and analyzed during the current study are available from the author Shu Haowen upon reasonable request but are not yet publicly available due to ongoing research.

Code Availability

Not applicable.

Authors' Contributions

Xin Yu, Guoliang Ren, are responsible for designing the framework, analyzing the performance, validating the results, and writing the study. Jie Niu is responsible for collecting the information required for the framework, provision of software, critical review, and administering the process.

REFERENCES

[1] B. Sizirici, Y. Fseha, C.-S. Cho, I. Yildiz, and Y.-J. Byon, "A review of carbon footprint reduction in construction industry, from design to operation," *Materials*, vol. 14, no. 20, Art. no. 20, Jan. 2021, doi: 10.3390/ma14206094.

[2] Y. H. Labaran, V. S. Mathur, S. U. Muhammad, and A. A. Musa, "Carbon footprint management: A review of construction industry," *Cleaner Engineering and Technology*, vol. 9, p. 100531, Aug. 2022, doi: 10.1016/j.clet.2022.100531.

[3] G. Wang et al., "A comprehensive review of building lifecycle carbon emissions and reduction approaches," *City and Built Environment*, vol. 2, no. 1, p. 12, Oct. 2024, doi: 10.1007/s44213-024-00036-1.

[4] L. Chen et al., "Green construction for low-carbon cities: A review," *Environmental Chemistry Letters*, vol. 21, no. 3, pp. 1627–1657, Jun. 2023, doi: 10.1007/s10311-022-01544-4.

[5] A. Fnais et al., "The application of life cycle assessment in buildings: Challenges, and directions for future research," *International Journal of Life Cycle Assessment*, vol. 27, no. 5, pp. 627–654, May 2022, doi: 10.1007/s11367-022-02058-5.

[6] Y. Li et al., "A review on the policy, technology and evaluation method of low-carbon buildings and communities," *Energies*, vol. 16, no. 4, Art. no. 4, Jan. 2023, doi: 10.3390/en16041773.

[7] Z. Chen et al., "Recent technological advancements in BIM and LCA integration for sustainable construction: A review," *Sustainability*, vol. 16, no. 3, Art. no. 3, Jan. 2024, doi: 10.3390/su16031340.

[8] A. Yang, M. Han, Q. Zeng, and Y. Sun, "Adopting building information modeling (BIM) for the development of smart buildings: A review of enabling applications and challenges," *Advances in Civil Engineering*, vol. 2021, no. 1, p. 8811476, 2021, doi: 10.1155/2021/8811476.

[9] B. P. Arsecularatne, N. Rodrigo, and R. Chang, "Review of reducing energy consumption and carbon emissions through digital twin in built environment," *Journal of Building Engineering*, vol. 98, p. 111150, Dec. 2024, doi: 10.1016/j.jobe.2024.111150.

[10] M. Alhammad, M. Eames, and R. Vinai, "Enhancing building energy efficiency through building information modeling (BIM) and building energy modeling (BEM) integration: A systematic review," *Buildings*, vol. 14, no. 3, Art. no. 3, Mar. 2024, doi: 10.3390/buildings14030581.

[11] S. Jayanthi, R. Lakshmana Kumar, P. Punitha, B. Muthu, and C. B. Sivaparthipan, "Sustainable energy harvesting techniques for underwater aquatic systems with multi-source and low-energy solutions," *Sustainable Computing: Informatics and Systems*, vol. 46, p. 101126, 2025. doi: 10.1016/j.suscom.2025.101126.

[12] X. Chen and X. Chen, "Data visualization in smart grid and low-carbon energy systems: A review," *International Transactions on Electrical Energy Systems*, vol. 31, no. 7, p. e12889, 2021, doi: 10.1002/2050-7038.12889.

[13] C. Xi and S.-J. Cao, "Challenges and future development paths of low carbon building design: A review," *Buildings*, vol. 12, no. 2, Art. no. 2, Feb. 2022, doi: 10.3390/buildings12020163.

[14] C. Z. Li, Y. Deng, Y. Ya, V. W. Y. Tam, and C. Lu, "Applications of information technology in building carbon flow," *Sustainability*, vol. 15, no. 23, Art. no. 23, Jan. 2023, doi: 10.3390/su152316522.

[15] A. AlJaber, E. Alasmari, P. Martinez-Vazquez, and C. Baniotopoulos, "Life cycle cost in circular economy of buildings by applying building information modeling (BIM): A state of the art," *Buildings*, vol. 13, no. 7, Art. no. 7, Jul. 2023, doi: 10.3390/buildings13071858.

[16] B. Mishra, T. Shanmugapriya, S.-H. Hsieh, Y.-T. Chang, and A. Pal, "BIM-based approach for analyzing carbon emissions of residential buildings in India," *Innovative Infrastructure Solutions*, vol. 10, no. 4, p. 153, Mar. 2025, doi: 10.1007/s41062-025-01950-x.

[17] N. Fonseca Arenas and M. Shafique, "Recent progress on BIM-based sustainable buildings: State of the art review," *Developments in the Built Environment*, vol. 15, p. 100176, Oct. 2023, doi: 10.1016/j.dibe.2023.100176.

[18] C.-L. Ho, C.-C. Wang, S. Qi, and Z. Zhang, "Data-driven optimization for low-carbon prefabricated components production based on ant colony algorithms," *Buildings*, vol. 14, no. 12, Art. no. 12, Dec. 2024, doi: 10.3390/buildings14124060.

[19] N. Razi and R. Ansari, "A prediction-based model to optimize construction programs: Considering time, cost, energy consumption, and CO2 emissions trade-off," *Journal of Cleaner Production*, vol. 445, p. 141164, Mar. 2024, doi: 10.1016/j.jclepro.2024.141164.

[20] A. Popa, A. P. Ramallo González, G. Jaglan, and A. Fensel, "A semantically data-driven classification framework for energy consumption in buildings," *Energies*, vol. 15, no. 9, p. 3155, Apr. 2022, doi: 10.3390/en15093155.

[21] [N. N. Myint and M. Shafique, "Embodied carbon emissions of buildings: Taking a step towards net zero buildings," *Case Studies in Construction Materials*, vol. 20, p. e03024, Jul. 2024, doi: 10.1016/j.cscm.2024.e03024.

[22] L. Zhao, H. Zhang, Q. Wang, and H. Wang, "Digital-twin-based evaluation of nearly zero-energy building for existing buildings based on scan-to-BIM," *Advances in Civil Engineering*, vol. 2021, no. 1, p. 6638897, 2021, doi: 10.1155/2021/6638897.

[23] K. Shen, L. Ding, and C. C. Wang, "Development of a framework to support whole-life-cycle net-zero-carbon buildings through integration of building information modelling and digital twins," *Buildings*, vol. 12, no. 10, Art. no. 10, Oct. 2022, doi: 10.3390/buildings12101747.

[24] F. Tahmasebinia et al., "Using regression model to develop green building energy simulation by BIM tools," *Sustainability*, vol. 14, no. 10, p. 6262, May 2022, doi: 10.3390/su14106262.

[25] D. Zhuang et al., "A performance data integrated BIM framework for building life-cycle energy efficiency and environmental optimization design," *Automation in Construction*, vol. 127, p. 103712, Jul. 2021, doi: 10.1016/j.autcon.2021.103712.

[26] M. H. Mehraban, A. A. Alnaser, and S. M. E. Sepasgozar, "Building information modeling and AI algorithms for optimizing energy performance in hot climates: A comparative study of Riyadh and Dubai," *Buildings*, vol. 14, no. 9, Art. no. 9, Sep. 2024, doi: 10.3390/buildings14092748.

[27] H. Wang et al., "Integrating BIM and machine learning to predict carbon emissions under foundation materialization stage: Case study of China's 35 public buildings," *Frontiers of Architectural Research*, vol. 13, no. 4, pp. 876–894, Aug. 2024, doi: 10.1016/j.foar.2024.02.008.

[28] A. Arsiwala, F. Elghaish, and M. Zoher, "Digital twin with machine learning for predictive monitoring of CO2 equivalent from existing buildings," *Energy and Buildings*, vol. 284, p. 112851, Apr. 2023, doi: 10.1016/j.enbuild.2023.112851.

[29] S. Giannelos, F. Bellizio, G. Strbac, and T. Zhang, "Machine learning approaches for predictions of CO2 emissions in the building sector," *Electric Power Systems Research*, vol. 235, p. 110735, Oct. 2024, doi: 10.1016/j.epsr.2024.110735.

[30] A. Mohammed Alshehri et al., "Building information modeling (BIM) driven performance-based construction for the optimization of sustainable and smart structures development," *Environmental Challenges*, vol. 16, p. 100980, Aug. 2024, doi: 10.1016/j.envc.2024.100980.

[31] P. Schneider-Marin, H. Harter, K. Tkachuk, and W. Lang, "Uncertainty analysis of embedded energy and greenhouse gas emissions using BIM in early design stages," *Sustainability*, vol. 12, no. 7, Art. no. 7, Jan. 2020, doi: 10.3390/su12072633.

[32] L. A. van Ellen, B. N. Bridgens, N. Burford, and O. Heidrich, "Rhythmic buildings - a framework for sustainable adaptable architecture," *Building and Environment*, vol. 203, p. 108068, Oct. 2021, doi: 10.1016/j.buildenv.2021.108068.

[33] R. Huang and S. Mao, "Carbon footprint management in global supply chains: A data-driven approach utilizing artificial intelligence algorithms," *IEEE Access*, vol. 12, pp. 89957–89967, 2024, doi: 10.1109/ACCESS.2024.3407839.

[34] M. Arishi and M. Kuku, "Mitigating carbon emissions through AI-driven optimization of zeolite structures: A hybrid model approach," *Alexandria Engineering Journal*, vol. 115, pp. 370–389, Mar. 2025, doi: 10.1016/j.aej.2024.12.049.

[35] Y. Natarajan, S. K. Saha, S. Karunakaran, R. Sundararajan, and G. Dhiman, "Enhancing building energy efficiency with IoT-driven hybrid deep learning models for accurate energy consumption prediction," *Sustainability*, vol. 16, no. 5, art. no. 1925, Jan. 2024. doi: 10.3390/su16051925.

[36] M. Symeonides, N. Tsiopani, G. Maouris, D. Trihinas, G. Pallis, and M. D. Dikaiakos, "CarbonOracle: Automating energy mix & renewable energy source forecast modeling for carbon-aware micro data centers," in *Proc. IEEE/ACM 17th Int. Conf. Utility and Cloud Comput. (UCC)*, Dec. 2024, pp. 246–255. doi: 10.1109/UCC63386.2024.00042.

JellyNovaNet-JSO: A Hybrid TabNet–BiLSTM Model for IoT-Based Crop Yield Prediction

Huang Zhicheng*, Zhang Yinjun
Guangxi Science and Technology Normal University, China

***Abstract*—Precise prediction of crop yield is essential for sustainable agriculture, resource maximization, and food security. As the use of IoT and Wireless Sensor Networks (WSNs) gains momentum, huge amounts of heterogeneous and time-series environmental data have become readily available from intelligent greenhouses. Despite this, it is still difficult to obtain meaningful insights from these data due to their high dimensionality, noise, and nonlinear temporal behavior. Traditional machine learning and statistical approaches usually fail to effectively capture static as well as sequential relationships, and most current models are difficult to tune hyperparameters and have problems with dealing with data heterogeneity and do not generalize across dynamic environments. To overcome these shortcomings, this paper introduces JellyNovaNet-JSO, a new hybrid deep learning architecture that integrates TabNet and BiLSTM architectures, designed using the Jellyfish Search Optimization (JSO) algorithm. The model exploits TabNet sparse attention for static feature modeling and the temporal memory of BiLSTM for time-series sensor data. The innovation is in utilizing attention-guided tabular learning with bidirectional temporal modeling, with a metaheuristic optimization layer to perform automatic hyperparameter tuning. Experimental outcomes based on real-world IoT greenhouse data demonstrate that JellyNovaNet-JSO attains MAE of 0.012, RMSE of 0.017, R^2 of 0.991, and MAPE of 1.89%, outperforming state-of-the-art CNN-LSTM, Random Forest, and SVM models substantially. In comparison with the prior approaches, JellyNovaNet-JSO enhances prediction accuracy by as much as 25% while ensuring scalability and robustness. This innovation provides a viable, interpretable, and deployable solution for precision agriculture, enabling smarter irrigation, climate control, and yield management.**

***Keywords*—IoT agriculture; crop yield prediction; BiLSTM; TabNet; jellyfish search optimization**

I. Introduction

Accurate crop yield estimation is a key consideration in modern agriculture, as a valuable decision-making aid for farmers, policymakers, and stakeholders to rely on food production [1], resource deployment, and planning for marketing [2]. With the world's population increasing and agricultural land shrinking, crop yield maximization is needed to ensure food security and sustainable agricultural output [3]. Sufficient and reliable yield estimates enable risk avoidance of adverse weather conditions, infestation by pests, and other environmental factors that could seriously hinder agricultural output [4]. Furthermore, authentic predictions enable proper input management at minimal cost in the form of water, fertilizers, and pesticides, thereby reducing costs as well as environmental deterioration [5]. Advances in technology, especially the integration of Internet of Things (IoT) devices and Wireless Sensor Networks (WSNs), have revolutionized the ability to monitor real-time environmental parameters across large agricultural fields [6]. Such a proliferation of data availability, though, comes with the difficulty of extracting useful information from high-dimensional, time-series, and intricate data [3]. Hence, it becomes imperative to develop robust predictive models that can accommodate heterogeneous types of data and temporal patterns [5]. These models not only facilitate proactive crop management but also propel sustainable agrarian practice by optimizing the utilization of resources and minimizing wastage [7]. Therefore, enhancing the accuracy and reliability of crop yield prediction models remains a top agenda for agriculture research, driving developments in machine learning [8], deep learning, and optimization techniques tailored to the unique demands of smart agriculture [6].

While great potential of IoT and Wireless Sensor Networks resides in agriculture, accurate estimation of crop yields is stifled by numerous significant challenges [1]. One of the significant challenges is the complexity and heterogeneity of the environmental data perceived by the different sensors monitoring variables such as temperature, humidity, soil nutrient concentrations, and water levels [9]. They are likely to exhibit nonlinear relationships and interact with each other in intricate ways that are difficult to model with traditional statistical techniques [10]. Furthermore, the data tend to be high-dimensional and noisy and contain missing entries and measurement errors due to sensor imperfections or environmental perturbations [11]. Another principal challenge is to model temporal dependencies and dynamic patterns inherent in agriculture processes [1]. Crop development and yields are contingent upon time-series fluctuations in environmental variables, so models need to be capable of learning long and short-term dependencies in sequential data [12]. Seasonal variations, soil nutrient lag effects, and periodic irrigation cycles make the time-scale modeling task even more complicated. In addition, the integration of static tabular information (e.g., crop type, soil type) with sequential sensor data contributes to the difficulties encountered by predictive modeling [9]. These challenges call for advanced machine learning paradigms with the capacity to deal with heterogeneous time-dependent information that also demands interpretability and robustness [10]. Overcoming these challenges is central in the development of plausible crop yield prediction systems deployable in real-world smart farming contexts [13].

The emergence of the IoT and WSNs [9] has revolutionized the field of agriculture through continuous, real-time

*Corresponding Author

monitoring of environmental parameters with unparalleled detail [14]. IoT is a network of physical objects with sensors, actuators, and communication modules which are connected with each other and communicate with the help of the internet or local networks [15]. In farming, IoT sensors track vital parameters including soil moisture, temperature, humidity, light levels, and nutrient content, and present farmers with accurate data to facilitate optimal crop care[16]. Additionally, WSN are spatially distributed nodes of sensors that wirelessly interface to a central gateway, allowing for flexible and extensible data collection in large-scale farms or contained environments such as greenhouses[17]. WSNs have the benefit of simple deployment, low power requirements, and environmental challenge resistance, thus being well suited for agricultural monitoring[18]. The combination of IoT and WSN technologies produces rich, multivariate, and time-stamped data that represents dynamic interactions between crop and environment[18], [19]. This richness in data provides the foundation for building advanced predictive models to enhance yield forecasting, resource management, and sustainable agriculture[20], [21], [36].

To address the issues concerned with crop yield prediction from heterogeneous and time-series farm data, this research presents JellyNovaNet-JSO, a new hybrid model that combines synergistically the merits of TabNet and Bidirectional Long Short-Term Memory (BiLSTM) networks, fine-tuned by the Jellyfish Search Algorithm (JSA). TabNet, being interpretable and having the capacity to effectively process tabular data, excels at capturing significant features from static or non-temporal environmental conditions such as water levels and soil moisture. On top of that, BiLSTM captures temporal dependencies and bidirectional contextual information from sequential sensor inputs such as temperature and humidity across time, which are crucial in modeling crop growth dynamics. The combination of these two architectures enables the model to utilize both static and sequential data modalities in an optimal manner and hence improves prediction accuracy. To further optimize model performance, the Jellyfish Search Algorithm as a bio-inspired metaheuristic search approach is employed in tuning major hyperparameters such as learning rates, layer dimensions, and batch setup. This optimization not only improves convergence but also avoids local minima, resulting in a robust and reliable crop yield prediction model ready for use in real-world smart agriculture scenarios.

This research makes the following key contributions:

1) Introduced a novel hybrid model, JellyNovaNet-JSO, which combines the TabNet architecture with BiLSTM networks. The model is optimized using the Jellyfish Search Algorithm (JSO) to improve the accuracy and efficiency of crop yield prediction in IoT-enabled greenhouse environments.

2) Leveraged real-time data collected from IoT-based sensors and wireless sensor networks deployed in a smart greenhouse to model crop growth patterns and predict yield.

3) Incorporated advanced data preprocessing techniques, including timestamp conversion, feature scaling, and temporal feature engineering. These methods enhance the model's ability to learn from both static and temporal features, improving prediction accuracy. The BiLSTM component of the proposed hybrid model captures bidirectional temporal dependencies in time-series sensor data. This is essential for modeling complex, long-term, and short-term crop growth patterns, which are influenced by changing environmental conditions.

4) Conducted a thorough estimation of the model using several performance metrics, including MSE and R^2, to assess the accuracy of crop yield predictions. The approach is validated through the prediction of crop yields in a real-world IoT-enabled greenhouse setup.

The rest of this paper is structured as follows:

- Section II summarizes recent publications on IoT and ML/DL-based crop yield prediction, citing current drawbacks.
- Section III formulates the problem statement and states the main goals encompassing scalability, heterogeneity of data, adaptability, and tuning.
- Section IV presents the envisioned JellyNovaNet-JSO hybrid model, including its architecture, elements (TabNet, BiLSTM), preprocessing pipeline, and JSO optimization.
- Section V offers results of experiments, such as evaluation measures, attention analysis, ablation analysis, and performance comparisons with other models.
- Section VI summarizes the study and proposes directions for future work such as real-time deployment and multi-location validation.

II. Related Works

N. Chandiraprakash et al. [14] created a crop yield forecasting model that combined real-time IoT sensor data with state-of-the-art ML algorithms, such as LSTM and Random Forest, to facilitate better adaptation to climatic variability. They also investigated the integration of CNN with LSTM to achieve higher spatial-temporal prediction accuracy. They had a user-friendly interface as part of actionable farmer insights for precision agriculture. Rath et al. [22] suggested a precision farming methodology based on the use of real-time wireless sensor network data for enhanced accuracy in crop yield forecasting. Their model incorporates sensor feeds on a continuous basis through a mobile app, providing timely prediction and real-time data gathering capabilities. The research discusses the advantages of utilizing sensor-assembled data from many users for improving model training. Haritha et al. [23] worked on a supervised machine learning method of crop prediction based on climatic and soil factors like humidity, temperature, and nutrient levels. They tested various classifiers like Naïve Bayes, AdaBoost, Decision Tree, and Voting Classifier and determined that the Decision Tree algorithm ranked highest with 99.4% accuracy. The work is centered on enhancing prediction accuracy to help farmers in selecting crops in fluctuating climatic conditions. Çetiner et al. [3] suggested a hybrid deep model of LSTM and CNN to automatically predict crop yields with major parameters like water consumption, exposure to sunlight, fertilizers, pesticides, and fields of cultivation. Their model provided good predictive

power with an R^2 measure of 89.71 along with low error metrics, establishing competitiveness for state-of-the-art methods.

Prathap et al. [24] investigated an IoT-based intelligent agriculture system to enhance crop yield prediction and automated monitoring to minimize human involvement. Their method combines sensors and cameras to monitor yield levels, uploading real-time data to the cloud for analysis. A hybrid deep learning algorithm is used in the study to improve prediction accuracy in smart agriculture systems. Sensor reliability and integration of data issues still exist for fully automated systems. Gupta and Nahar et al. [25] created a hybrid machine learning system with IoT data for crop yield prediction, incorporating preprocessing, feature extraction, and classification steps. The two-level approach utilizes aKNCN for classifying soil quality and ELM with an improved BOA for predicting yields. The model revealed better accuracy and reduced errors on soil datasets with varied evaluation measures. Manikandababu et al. [26] examined how machine learning algorithms and IoT technologies can be integrated to improve crop production prediction in precision agriculture. They employed sensor and drone data measuring crop health, weather, and soil moisture for informing models. This method allows farmers to make informed decisions regarding pest control, fertilization, and irrigation, increasing yield accuracy. N. Mohana Priya et al. [27] conducted a study on the combination of ML and IoT for the optimization of irrigation management based on real-time sensor readings of temperature, humidity, soil moisture, and water level. They compared models with SVM resulting in the highest accuracy of pump operation prediction. The study showcases the capability of IoT and ML to make adaptive, data-driven control of irrigation for better water use efficiency. Pérez et al. [28] created a tomato crop forecasting system combining AI, sensor networks and IoT to balance resource utilization and improve forecasting yield. The framework included distributed sensors, IoT gateways, and cloud-based recurrent neural network models that were trained on environmental information and tested with harvest data. Their model had an average prediction error of 3.2% over a period of four weeks, reflecting high accuracy.

Kumar et al. [29] implemented an ensemble ML model for crop prediction from IoT sensor data gathered using the PLX-DAQ tool, integrating algorithms. Their ensemble method has a very high accuracy of 97.45% for early crop yield prediction. Their work showcases the strength of an IoT and ML integration for maximizing data-driven decisions in agriculture amid the threats posed by climate change. Yet, the dependency on sensor integration and data quality continues to be a bottleneck in sweeping adoption. Krithika et al. [30] explored DL model applied for crop yield prediction based on real-time agricultural datasets. The experiments showed that although all models were promising, LSTM was most accurate because it can learn temporal dependencies in the data. The research underscores the importance of improving ML models further to improve attempts in mitigating challenges brought by natural disasters. Rastog et al. [31] proposed an AI and ML approach with Python to improve crop yield prediction and assist farmers in the Indian subcontinent by leveraging IoT and Cyber-Physical systems. Their research is intended to tackle food security issues by making timely predictions of yield, moisture, and weather conditions. The research endeavors to enhance prediction accuracy through sophisticated learning algorithms on authentic agricultural datasets.

Although recent work shows substantial advancements in using IoT and machine learning for crop yield estimation and precision farming, some universal limitations are still prevalent. Computational complexity and scalability issues of hybrid CNN-LSTM models were noted by Chandiraprakash et al. [14] in large-scale implementations. Rath et al. [22] mentioned issues with handling heterogeneous sensor data and maintaining model scalability across various agricultural environments. Haritha et al. [23] pointed out that models based on fixed datasets could face challenges in coping with dynamic climatic conditions. ÇetiNer et al. [3] highlighted the reliance on precise parameter measurement and rigorous fine-tuning for consistent model performance. Prathap et al. [24] indicated challenges with sensor reliability and data integration as challenges to complete automation. Gupta and Nahar et al. [25]found that the use of hyperparameter tuning would restrict generalizability in various contexts of farming. Manikandababu et al. [26] had mentioned difficulties in large-scale deployment and integration of data. Mohana Priya et al. [27] reported misclassification error in irrigation control predictions in need of further tuning. Pérez et al. [28] showed high accuracy but recognized the necessity for strong system integration. Kumar et al. [29] and Rastog et al. [31]pointed to bottlenecks in sensor quality data and scalability challenges for mass implementation. Taken together, the studies point toward the fact that in addition to much encouraging work, challenges like heterogeneity of data, computational costs, reliability of sensors, robustness of models, and scalability need to be overcome to achieve the complete potential of IoT and ML in smart agriculture.

III. Problem Statement

In spite of remarkable progress in unifying IoT and machine learning for crop yield prediction, some urgent limitations obviate widespread use and performance. First, most models are computationally expensive and lack scalability when applied in practical large-scale agricultural settings [14]. Second, data heterogeneity and sensor unreliability issues impair input data quality and uniformity, degrading model robustness [22]. Third, the reliance on fixed or static datasets lowers the models' responsiveness to dynamic and rapidly evolving environmental conditions [23]. Lastly, most current models are sensitive to hyperparameters that need to be heavily tuned and fine-tuned, affecting generalizability across a wide range of farming contexts [25]. Such constraints warrant more effective, adaptive, and scalable precision agriculture solutions.

Research Objectives

- Develop a computationally efficient hybrid model that maintains high accuracy while enabling scalability for large-scale agricultural deployments.
- Design robust data preprocessing and sensor fusion techniques to handle data heterogeneity and improve sensor data reliability.

- Incorporate dynamic learning mechanisms that adapt to changing environmental and climatic conditions in real-time.
- Implement an automated hyperparameter optimization framework to enhance model generalizability and reduce manual tuning efforts.

IV. Proposed Methodology for Crop Yield Prediction Using IoT and Wireless Sensor Networks

The methodology for crop yield prediction put forward uses a hybrid machine learning model, JellyNovaNet-JSO, that combines the TabNet and BiLSTM architectures, optimized using the JSO. The proposed methodology takes advantage of the robustness of both the TabNet model, specifically developed for effective processing of tabular and categorical data, and the BiLSTM model, which is able to recognize bidirectional temporal patterns in time-series data. The model initially handles static attributes such as soil nutrient content and actuator states through TabNet, whereas BiLSTM extracts the dynamic behavior of sequential sensor readings from IoT-connected greenhouses. These models are then combined to merge spatial and temporal information so that accurate and efficient crop yield prediction can be supported. With timestamped features, feature scaling, and time series windowing, the hybrid method offers comprehensive modeling of environmental circumstances and their impacts on crop growth while providing a more robust setting for yield prediction in smart agricultural applications. Fig. 1 demonstrates the step-by-step methodology for crop yield prediction using the JellyNovaNet-JSO hybrid model integrating IoT sensor data and machine learning techniques.

A. Data Collection

The dataset in the present research was obtained as part of research for the master's thesis carried out by Mohammed Ismail Lifta (2023-2024) at Tikrit University, Iraq [32]. The data was obtained from a mounted smart greenhouse that is equipped with advanced IoT-enabled sensors and actuators. Environmental conditions such as, humidity, temperature, water level, and soil nutrient concentrations (nitrogen, phosphorus, potassium) were sensed in real time through wireless sensor nodes placed around the greenhouse. Actuator status such as fan and pump ON/OFF signals were also recorded in order to express environmental control decisions. Data, spanning multiple months and comprising 37,923 records, were remotely transmitted and stored via a cloud-capable application linked to Google Sheets for real-time monitoring and regulation of the greenhouse climate. The high-density, time-stamped, multivariate data are most suitable for modeling crop growth patterns and yield prediction using IoT and Wireless Sensor Network technologies. This Table I shows exemplary data of time-stamped sensor measurements read from the smart greenhouse. It further depicts actuator statuses reflecting the state of operation of the fan and watering pumps, capturing environmental conditions as well as control actions. The data corresponds to tomatoes grown in the smart greenhouse for four continuous months in 2023-2024. During preprocessing, minor issues were noted and handled: missing entries, sensor drift, and dropped packets. The complete dataset is available for reproducibility at:

https://www.kaggle.com/datasets/wisam1985/iot-agriculture-2024

B. Data Preprocessing for JellyNovaNet-JSO Model

Some preprocessing steps need to be carried out after having a clean dataset with one-hot encoded values captured from the smart greenhouse sensor network before being modeled efficiently using the JellyNovaNet-JSO hybrid architecture. The main tasks include timestamp conversion, feature scaling, temporal feature engineering, time series windowing, and separating the dataset into training and testing so that data leakage does not occur.

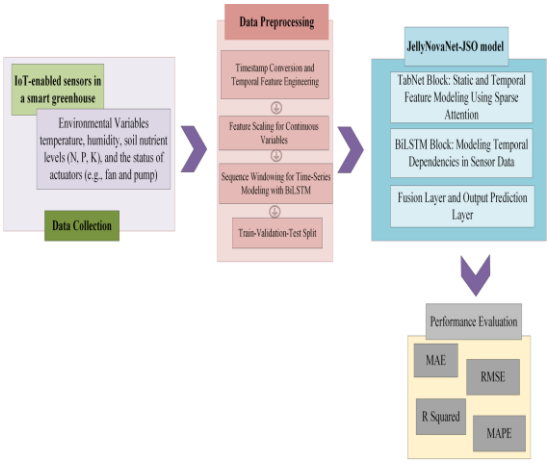


Fig. 1. Methodology flow for crop yield prediction using JellyNovaNet-JSO hybrid model.

TABLE I Sample Environmental and Actuator Sensor Data from Smart Greenhouse

Date & Time	Temperature (°C)	Humidity (%)	Water Level (%)	N (0-255)	P (0-255)	K (0-255)	Fan actuator ON	Watering plant pump ON	Water pump actuator ON
2024-02-08 06:10:00	41	63	100	255	255	255	1	1	1
2024-02-08 06:15:00	41	59	100	255	255	255	1	1	1
2024-02-08 06:20:00	41	62	100	255	255	255	1	1	1
2024-01-18 05:02:00	37	62	100	255	255	255	1	1	1
2024-01-18 05:07:00	37	63	100	255	255	255	1	1	1

1) Timestamp conversion and temporal feature engineering: The dataset contains a critical Date and Time column representing the exact timestamp for each sensor reading. To leverage temporal patterns and seasonality effects in crop growth, this column must be converted into a datetime data type compatible with modeling frameworks as given in Eq. (1)

$$\text{Timestamp}_i = \text{datetime(Date \& Time } i) \quad (1)$$

This conversion enables extraction of granular temporal features such as hour of day, day of month, month, and day of week, which can serve as additional predictors to improve model performance. These features are mathematically derived from the timestamp as given in Eq. (2).

$$\begin{aligned} \text{Hour}_i &= \text{Timestamp}_i\text{.hour} \\ \text{Day}_i &= \text{Timestamp}_i\text{.day} \\ \text{Month}_i &= \text{Timestamp}_i\text{.month} \\ \text{DayOfWeek}_i &= \text{Timestamp}_i\text{.weekday} \end{aligned} \quad (2)$$

Incorporating these gyclic temporal components allows the model to capture daily and seasonal variations inherent in environmental and crop growth data.

2) Feature scaling for continuous variables: The continuous sensor features, including temperature, humidity, water level, and soil nutrient concentrations (nitrogen, phosphorus, potassium), span different numerical ranges. To prevent features with larger scales from leading the learning process and to accelerate convergence during training, feature scaling is necessary. The Min-Max scaling technique normalizes these features to a fixed range [0,1], using Eq. (3).

$$x_i' = \frac{x_i - x_{min}}{x_{max} - x_{min}} \quad (3)$$

where x_i is the actual feature value, and $x_{\min}$, $x_{\max}$ are the min and max values of that feature across the dataset.

3) Sequence windowing for time-series modeling with BiLSTM: BiLSTM networks need sequential input data that capture temporal dependencies. To convert the continuous sensor data into appropriate time-series format, a sliding window approach was applied. With a defined window size ω, consecutive segments of the dataset were extracted as input sequences, as shown in Eq. (4).

$$S_t = [x_t, x_{t+1}, \dots, x_{t+w-1}] \quad (4)$$

where S_t is the sequence input at time step t and x_i is the feature vector at timestamp i.

The target label (e.g., crop yield or environmental condition at $+w$) can be associated with each sequence, enabling supervised learning. The choice of window size balances capturing long-term dependencies without making the sequence overly large, typically ranging from 5 to 30-time steps depending on data frequency.

4) Train-validation-test split: To simulate real-world forecasting and prevent data leakage, the dataset is split chronologically rather than randomly. The data are divided into three subsets, given in Eq. (5).

$$\begin{cases} \text{Training set} & : & \text{first } 70\% \\ \text{Validation set} & : & \text{next } 15\% \\ \text{Test set} & : & \text{final } 15\% \end{cases} \quad (5)$$

Such time-based splitting confirms that the model is tested on future unseen data, mimicking actual deployment scenarios in crop yield prediction.

The data preprocessing pipeline caters to the requirements of JellyNovaNet-JSO, that is, it transforms cleaned one-hot encoded raw data into an appropriate format. The steps involved are timestamp parsing, temporal feature generation, normalization of continuous variables, creation of sequential windows for BiLSTM input, and finally the time-aware train/test split. Thus, the hybrid model can indeed grasp the complexities involved in spatio-temporal patterns of an IoT-enabled greenhouse environment to finally provide correct crop yield estimations.

Pseudocode: Data Preprocessing for JellyNovaNet-JSO

Input: RawDataset.csv # IoT-based greenhouse sensor data

Output: Preprocessed training, validation, and test sets

BEGIN

1. Load Dataset

Dataset ← read_csv("RawDataset.csv")

2. Timestamp Conversion

For each record in Dataset:

Timestamp_i ← to_datetime (Dataset ["Date & Time"] [i])

3. Temporal Feature Engineering

For each Timestamp_i:

Hour_i ← Timestamp_i. hour

Day_i ← Timestamp_i.day

Month_i ← Timestamp_i. month

DayOfWeek_i ← Timestamp_i. weekday

Append Hour, Day, Month, DayOfWeek as new columns to Dataset

4. Feature Scaling using Min-Max Normalization

For each continuous feature f ∈ {Temperature, Humidity, WaterLevel, N, P, K}:

f_scaled ← (f - min(f)) / (max(f) - min(f))

Replace original features with scaled features

5. Sequence Windowing for Time-Series Input

Define window_size w

Sequences ← []

Labels ← []

For i from 0 to (length (Dataset) - w):

Sequence ← Dataset [i: i+w]

Label ← Dataset [i + w] ["Crop_Yield"]

Append Sequence to Sequences

Append Label to Labels

6. Chronological Train-Validation-Test Split

total_samples ← length (Sequences)

train_end ← 0.70 * total_samples

val_end ← 0.85 * total_samples

TrainSet ← Sequences

Train Labels ← Labels

ValSet ← Sequences

ValLabels ← Labels

TestSet ← Sequences

TestLabels ← Labels

7. Return TrainSet, ValSet, TestSet and corresponding labels

END

C. Model Development

To apply the JellyNovaNet-JSO model a Jellyfish Search Optimized variant of TabNet and BiLSTM for precise crop yield forecasting. The suggested JellyNovaNet-JSO model is a hybrid DL system combining the features of TabNet and BiLSTM networks, JSO-optimized to improve the precision of crop yield forecasting in IoT-based greenhouse setups. The model starts with a TabNet layer, which is particularly suited to process the heterogeneous tabular and categorical features like nutrient contents (N, P, K), actuator states (fan/pump ON/OFF), and engineered temporal variables (hour, day, month). TabNet applies a sparse attention mechanism to softly select relevant features at every decision step in order to make the model interpretable and learn efficiently from high-dimensional structured data. In parallel to this, the BiLSTM module handles the sequential input from time-series sensor reading data (e.g., water level, temperature, humidity) formatted into sliding windows. The BiLSTM can model intricate bidirectional temporal relationships, allowing the model to learn past and future contextual patterns affecting crop development. The features from TabNet and BiLSTM are then input into a layer of feature fusion, where they are concatenated and optionally smoothed by dense layers to facilitate cross-modal interaction between static tabular knowledge and dynamic temporal patterns. Lastly, the fused representation is input into a fully connected output layer that predicts the final continuous crop yield. This hybrid architecture guarantees that the model is efficiently able to capture the spatial and temporal complexities of IoT greenhouse data, while JSO integration further boosts the performance of the model using smart hyperparameter tuning.

1) TabNet block: Static and Temporal Feature Modeling Using Sparse Attention.

The TabNet block is used to process the tabular and categorical features of IoT sensor data and temporal encodings. They consist of static variables such as soil nutrient content (N, P, K), actuator states (fan ON/OFF, water pump state), and engineered time-based features like hour, day, month, and day of the week - all of which are pivotal in comprehending crop growth dynamics in smart greenhouses. Fig. 2 shows the TabNet architectural diagram and their components are explained briefly given below.

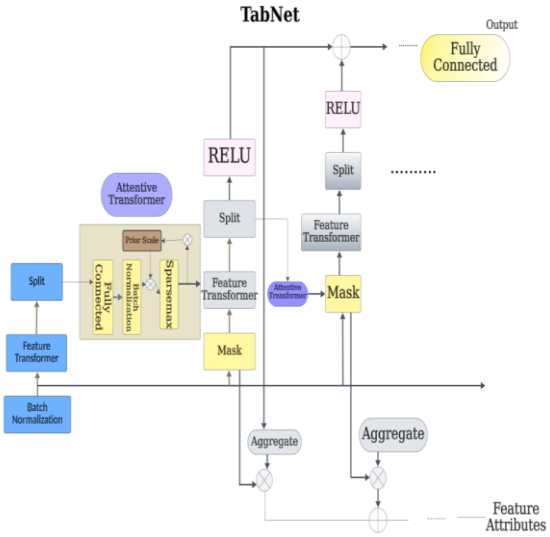


Fig. 2. TabNet architectural diagram.

Input Representation:

Let the input feature vector be represented as in Eq. (6).

$$X = \{x_1, x_2, \ldots, x_n\} \in \mathbb{R}^n \quad (6)$$

where each $\boldsymbol{x_i}$ corresponds to a scaled and encoded feature (e.g., N, P, K, actuator state, hour, day, etc.), and $\boldsymbol{n}$ is the entire number of tabular features.

a) Feature transformer: Each input is passed through a feature transformer, which contains fully connected layers, batch normalization, and non-linear activation functions. This layer learns complex feature interactions and generates a richer representation of the input. For a layer $\boldsymbol{l}$, the transformation can be described as in Eq. (7).

$$H^{(l)} = ReLU\left(BN\left(W^{(l)}H^{(l-1)} + b^{(l)}\right)\right) \quad (7)$$

Where, $W^{(l)}$ *and* $b^{(l)}$ are weights and biases at layer l, $BN(\cdot)$ denotes batch normalization, $H^{(0)} = X$

The output $H^{(L)}$ of the final transformer layer is used as the latent feature embedding for downstream fusion.

b) Sparse attentive masking: To focus on the most relevant features at each decision step, TabNet incorporates a sparse attention mechanism. This mechanism generates feature masks M_i using the softmax and sparsemax functions. For a decision step i, the mask is computed as using Eq. (8).

$$M_i = sparsemax\left(P_{i-1} \cdot H^{(i)}\right) \quad (8)$$

Here, $H^{(i)}$ is the transformer output at step $\boldsymbol{i}$, $\boldsymbol{P_{i-1}}$ is the prior scale that limits the reuse of features (to promote sparsity), sparsemax (-) encourages hard selection of features, improving interpretability.

The masked input passed to the next decision step is given by Eq. (9).

$$X_i = M_i \odot X \quad (9)$$

where $\odot$ denotes element-wise multiplication.

c) Decision and aggregation: TabNet processes multiple decision steps T, where each step outputs a partial decision D_i that contributes to the final output given in Eq. (10) and Eq. (11)

$$D_i = \phi(H_i), \text{ for } i = 1 \ldots T \quad (10)$$

$$Y_{\text{TabNet}} = \sum_{i=1}^{T} D_i \quad (11)$$

where ϕ is a decision layer (e.g., linear transformation), and Y_{TabNet} is the final embedding output of the TabNet block. The result is a high-level latent feature embedding $Y_{\text{TabNet}} \in \mathbb{R}^d$, where d is the output dimension. These embedding captures complex, high-level relationships between the input features and will be later fused with the BiLSTM output for final prediction.

This TabNet block plays a critical role in modeling the non-sequential, structured feature space using sparse attention, ensuring both efficiency and interpretability, which are highly valuable in agricultural and loT applications. The output is forwarded to the fusion layer, where it is combined with the sequential BiLSTM representation for comprehensive crop yield forecasting.

2) BiLSTM block: Modeling Temporal Dependencies in Sensor Data.

The BiLSTM block in JellyNovaNet-JSO is considered to process the time-series data collected from the IoT-based greenhouse environment, such as temperature, humidity, water level, and other sequential features. These sensor readings are inherently temporal and exhibit short-term and long-term dependencies, which are critical for accurately modeling plant physiological responses and predicting final crop yield. Let the multivariate time-series input for a given window be given in Eq. (12).

$$S_t = [x_t, x_{t+1}, \ldots, x_{t+w-1}] \quad (12)$$

Where, w is the window size, $x_i \in \mathbb{R}^j$ is the feature vector at time i, and f is the number of continuous sensor features (e.g., temperature, humidity, water level, N, P, K).

This sequence captures both the temporal variation and feature evolution over a fixed period.

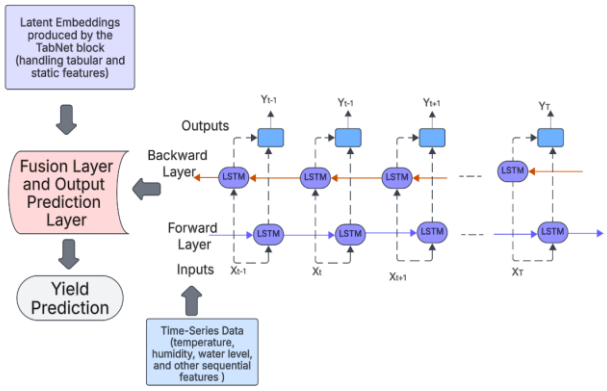


Fig. 3. BiLSTM- Architectural diagram.

Fig. 3 shows the architectural diagram of BiLSTM. Unlike traditional LSTM which processes sequences in a forward direction, BiLSTM reads input from both past and future directions, combining two LSTM outputs as given in Eq. (13).

$$h_t = \left[\overrightarrow{h_t}; \overleftarrow{h_t}\right] \quad (13)$$

Where, $\overrightarrow{h_t}$ is the hidden state from the forward LSTM (processing x_t to x_{t+w-1}), $\overleftarrow{h_t}$ is from the backward LSTM (processing x_{t+w-1} to x_t), [;] denotes concatenation.

Each LSTM unit at time step t operates with the following internal equations:
Let x_t be the input vector at time t, and let h_{t-1}, c_{t-1} be the earlier hidden and cell states. The gates and memory update are computed as using the following Eq. (14), (15), (16), (17), (18) and (19)

Let x_t be the input vector at time step t, and h_{t-1}, c_{t-1} represent the previous hidden state and cell state, respectively. The LSTM unit operates by regulating the flow of information using three gates: forget gate, input gate, and output gate, along with an internal cell state update mechanism.

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right)\text{(Forget gate)} \quad (14)$$

The forget gate f_t decides which information from the previous cell state should be discarded. It is computed using a sigmoid activation function applied to the linear transformation of the concatenated vector h_{t-1}, x_t, with associated weights W_f and bias b_f. The output is a vector of values between 0 and 1, indicating the degree of forgetting.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)\text{(Input gate)} \quad (15)$$

Which additional data should be added to the cell state is determined by the input gate. It makes use of a sigmoid activation, just like the forget gate. The same input is subjected to a tanh activation with weights and bias in parallel to create the candidate cell state. This candidate state is an example of possible new content that could be included.

$$\tilde{c}_t = tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \text{ (Candidate cell state)} \quad (16)$$

The new cell state is then updated by joining the scaled old cell state $\boldsymbol{f_t} \odot \boldsymbol{c_{t-1}} + \boldsymbol{i_t} \odot \boldsymbol{\tilde{c}_t}$ and the scaled candidate state $\boldsymbol{i_t} \odot \boldsymbol{\tilde{c}_t}$ denotes element-wise multiplication. This update mechanism allows the model to selectively retain or overwrite memory.

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \text{ (Updated cell state)} \quad (17)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)\text{(Output gate)} \quad (18)$$

$$h_t = o_t \odot tanh(c_t) \text{ (Hidden State)} \quad (19)$$

Where $\sigma(\cdot)$ is the sigmoid function, tanh(·) is the hyperbolic tangent function, O is the element-wise multiplication, W_* and b_* are weight matrices and biases for each gate. Both forward and backward passes compute their own set of hidden states, which are then concatenated to form a bi-directional context vector.

Temporal Embedding Vector

The BiLSTM processes the full sequence S_t and generates hidden states at each time step. These are often aggregated using:

- Last hidden state: h_{t+w-1}
- Mean pooling: $h_{\text{avg}} = \frac{1}{w}\sum_{i=t}^{t+w-1} h_i$
- Attention mechanism

The resulting temporal feature embedding $H_{\text{BiLSTM}} \in \mathbb{R}^d$ captures the dynamics and dependencies in the greenhouse environment over time. The BiLSTM block transforms sliding sequences of time-series sensor data into high-level temporal embeddings, effectively capturing both short- and long-term dependencies and bidirectional influences on crop behavior. This temporal representation H_{BiLSTM} is later fused with the TabNet output in the next stage for unified modeling and yield prediction.

3) Fusion layer and output prediction layer: The Fusion Layer in the JellyNovaNet-JSO architecture is responsible for combining the latent embeddings produced by the TabNet block (handling tabular and static features) and the BiLSTM block (handling timeseries sensor sequences). This fusion enables the model to integrate spatial, categorical, and temporal dependencies to make an informed prediction about crop yield. The outputs of the TabNet and BiLSTM are concatenated to create a single feature vector that incorporates both static and temporal dependencies. This fused representation is projected through one or more dense layers with relaxed linear unit (ReLU) activation which help smooth interactions of variables while accentuating cross-modal interactions. Lastly, a linear output layer will map this enriched representation into the predicted crop yield, using both static tabular features and continuous temporal dynamics of the sensor demand to produce next-harvest predictions.

a) Fusion layer: Concatenation of Latent Representations

Let, $z_{\text{TabNet}} \in \mathbb{R}^{d_1}$ be the feature embedding output from the TabNet branch. $z_{\text{BILSTM}} \in \mathbb{R}^{d_2}$ be the temporal embedding from the BiLSTM block.

These two vectors are concatenated into a unified feature representation, given in Eq. (20).

$$z_{\text{fused}} = \left[z_{\text{TabNet}}\,; z_{\text{BILSTM}}\right] \in \mathbb{R}^{d_1+d_2} \quad (20)$$

This fused vector contains both: Static environmental context (e.g., nutrient levels, actuator states, timestamp-derived features) and Temporal dynamics (e.g., historical patterns in temperature, humidity, water content, etc.).

b) Fully connected output layer: Yield Prediction: The fused vector is passed over one or more fully connected (dense) layers with non-linear activations (typically ReLU), followed by a final linear layer to produce the crop yield prediction, given in Eq. (21) and Eq. (22)

$$h_1 = ReLU\left(W_1 z_{\text{fused}} + b_1\right) \quad (21)$$

$$\hat{y} = W_2 h_1 + b_2 \quad (22)$$

Where, W_1, W_2 are the weights of the dense layers, b_1, b_2 are the corresponding biases, $\hat{y}$ is the predicted crop yield output (a scalar for regression).

Since crop yield prediction is a regression problem, the model is trained by minimalizing the MSE between the predicted and actual yield values, using Eq. (23).

$$\mathcal{L}_{MSE} = \frac{1}{N}\sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \quad (23)$$

Where, y_i is the actual crop yield for the i^{th} example, $\hat{y}_i$ is the predicted yield, N is the total number of training samples.

To prevent overfitting during training, additional techniques such as L2 regularization, Dropout, or Batch Normalization can be applied in the dense layers. The Fusion Layer integrates TabNet and BiLSTM representations, and the Output Layer translates this fused knowledge into an accurate prediction of crop yield. This stage finalizes the learning pipeline of the JellyNovaNet-JSO model and enables it to reason over both spatial and temporal cues in the environment.

Pseudocode: JellyNovaNet-JSO Model Development
Input:

X_tabular ← Tabular input features (N, P, K, actuator states, timestamp features)

X_sequence ← Time-series input (e.g., temperature, humidity, water level)

y_true ← Ground truth crop yield

Output:

y_pred ← Predicted crop yield

BEGIN

1. Initialize TabNet parameters (weights, masks)

2. For each decision step i = 1 to T:

H_i ← FeatureTransformer(H_{i-1})

M_i ← SparseMax(P_{i-1} · H_i)

X_i ← M_i ⊙ X_tabular

D_i ← DecisionLayer(H_i)

3. Y_TabNet ← Sum of all D_i (i = 1 to T)

4. For each sample in X_sequence:

Generate sequence window S_t of size w

5. Pass S_t through BiLSTM:

Forward_LSTM → → →

Backward_LSTM ← ← ←

6. Concatenate forward and backward hidden states:

h_t ← [h_t_forward; h_t_backward]

7. Temporal Embedding ← MeanPooling or LastHiddenState(h_t)

8. fused ← Concatenate (TabNet, Temporal Embedding)

9. h1 ← ReLU (W1 · z_fused + b1)

10. y_pred ← W2 · h1 + b2

11. Compute Loss (Mean Squared Error):

12. Apply backpropagation to minimize L_MSE

13. Use JSO to tune:

- Learning rate
- TabNet decision steps
- BiLSTM units
- Dropout rate
- Batch size

END

V. Results and Discussion

The JellyNovaNet-JSO model was implemented using Python 3.10 with TensorFlow and PyTorch backend on a system with an NVIDIA RTX 3080 GPU, 32 GB RAM. Key hyperparameters such as learning rate, window size, and number of decision steps in TabNet were optimized using the JSO algorithm. Table II shows the optimal hyperparameter values selected for the JellyNovaNet-JSO model. These values were fine-tuned to achieve the best performance in predicting crop yield.

TABLE II Optimal Hyperparameter Values for JellyNovaNet-JSO Model

Hyperparameter	Optimal Value
Learning Rate	0.0025
TabNet Decision Steps	5
BiLSTM Units	128
Batch Size	64
Dropout Rate	0.3
Window Sizea	10

A. Evaluation Metrics

To measure the prediction performance of the JellyNovaNet-JSO model in a real-world smart greenhouse environment using IoT and WSN data, the following evaluation metrics were applied:

1) MAE: In this research, a low MAE indicates that the JellyNovaNet-JSO model makes very small average errors, showcasing its ability to generalize well to unseen yield values across environmental conditions, expressed in Eq. (24).

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i| \quad (24)$$

2) RMSE: RMSE emphasizes that the model not only maintains small average errors but also avoids large outliers-critical for yield-sensitive decisions in automated greenhouses.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2} \quad (25)$$

R^2 Score

Indicates the proportion of the variance in the dependent variable that is foreseeable from the independent variables.

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2} \quad (26)$$

3) MAPE: Expresses prediction accuracy as a percentage, making it easier to interpret relative performance. A MAPE value demonstrates high reliability of JellyNovaNet-JSO for operational use, where even small forecasting errors can affect resource allocation and yield estimation.

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n} \left|\frac{y_i - \hat{y}_i}{y_i}\right| \quad (27)$$

After fine-tuning using the JSO, the JellyNovaNet-JSO model achieved outstanding predictive accuracy on the test set.

TABLE III MODEL EVALUATION METRICS

Metric	Value
MAE	0.012
RMSE	0.017
R² Score	0.991
MAPE	1.89%

Table III shows the model evaluation metrics, these indicate that the model arrests almost 99.1% of yield variance with exceptionally low prediction errors, making it viable for real-world smart farming deployment.

With little difference between expected and actual values, the model's accuracy and efficacy in predicting crop productivity show its dependability for practical precision agriculture applications. The JellyNovaNet-JSO model's anticipated crop yields and the actual observed yields are contrasted in Fig. 4.

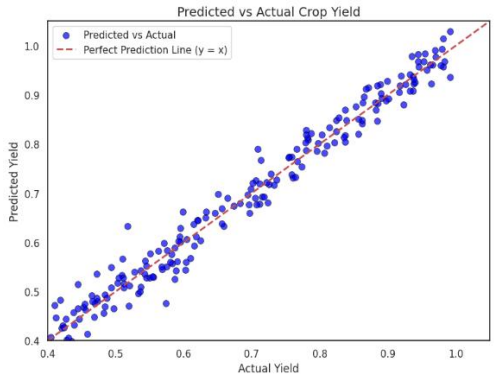


Fig. 4. Predicted vs. Actual crop yield.

The distribution of prediction errors (Actual – Predicted) for crop yield estimation shown in Fig. 5. The histogram is sharply centered around zero, indicating that the JellyNovaNet-JSO model maintains minimal error variance. This supports the model's low MAE and RMSE values and confirms its robustness for accurate yield forecasting in smart greenhouse environments.

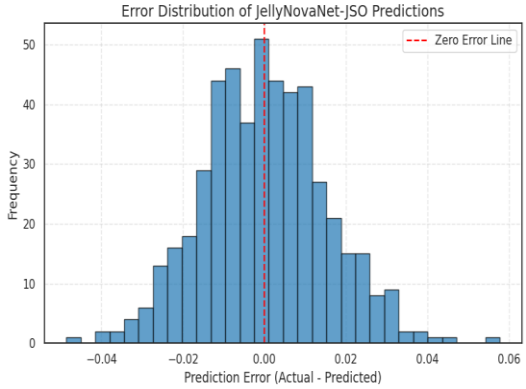


Fig. 5. Histogram of prediction errors for JellyNovaNet-JSO model.

Fig. 6 illustrates the model's loss over 20 training epochs. The training and validation losses converge smoothly without significant divergence, indicating stable learning. The absence of overfitting validates the use of a 0.3 dropout rate and effective hyperparameter tuning via the JSO, contributing to the model's generalization performance.

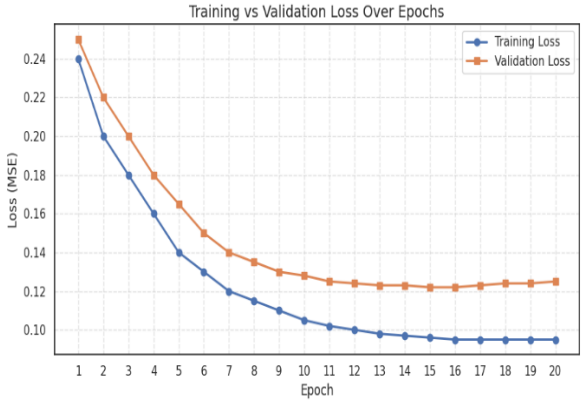


Fig. 6. Training and validation loss curve of JellyNovaNet-JSO model.

4) Feature importance from TabNet attention: Nutrient and humidity sensors had the strongest influence. Actuator signals also contributed to prediction by contextualizing environment control states. Table IV presents the feature importance scores (IoT + Actuator Features), and Fig. 7 illustrates them.

TABLE IV FEATURE IMPORTANCE SCORES (IOT + ACTUATOR FEATURES)

Feature	Importance Score
Nitrogen (N)	0.215
Humidity	0.192
Temperature	0.177
Water Level	0.151
Phosphorus (P)	0.116
Potassium (K)	0.099
Fan Actuator State	0.026
Pump Actuator State	0.024

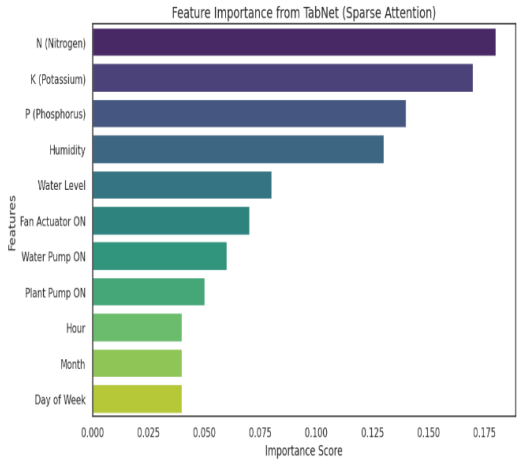


Fig. 7. Feature importance vs. Score.

5) BiLSTM temporal impact analysis: BiLSTM effectively captured greenhouse microclimate rhythms, especially temperature and water level shifts during day/night cycles. Table V showing average attention scores for time-series features.

TABLE V BILSTM TEMPORAL CONTRIBUTION SCORES

Sensor	Avg BiLSTM Score
Temperature	0.301
Humidity	0.283
Water Level	0.208
NPK Time Patterns	0.208

This Fig. 8 presents the average contribution scores of key time-series sensor features learned by the BiLSTM component. Temperature and humidity were identified as the most influential features, followed by water level and NPK time patterns. The visualization approves that BiLSTM effectively captures greenhouse microclimate rhythms, emphasizing its role in modeling temporal dependencies critical to correct yield prediction.

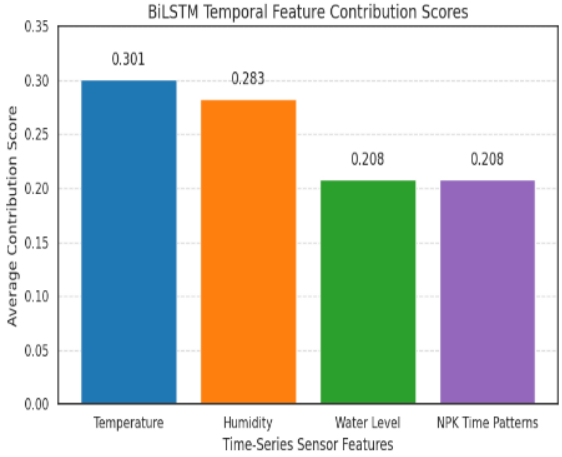


Fig. 8. BiLSTM temporal contribution scores for sensor features.

Fig. 9 shows the variation in predicted crop yields over actual crop yields year by year. This figure also illustrates the model's performance for different years, showing the ability of the JSO JellyNovaNet model to respond to altering environmental conditions and forecast crop yield accurately. The prediction was close to actual yields, providing evidence that the model can capture temporal dependencies and growth trends in crops with changes over time, serving real-time agricultural decision-making effectively.

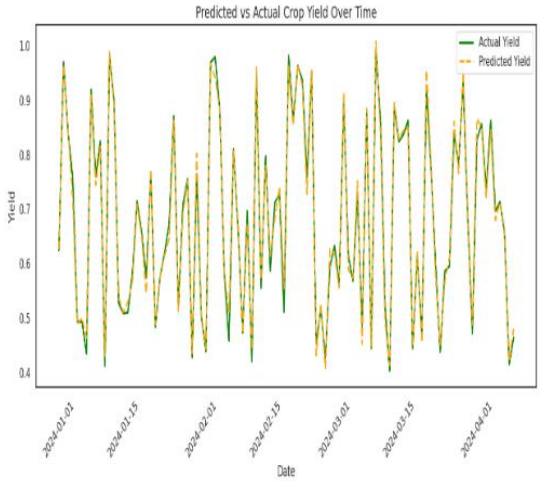


Fig. 9. Predicted vs. Actual crop yield over time.

This Fig. 10 demonstrates the normalized trends of key environmental sensor features temperature, humidity, and water level plotted alongside the actual crop yield across a 14-day period in January 2024. The left Y-axis denotes the normalized values of the sensor features, while the right Y-axis shows actual yield in kilograms. The visualization highlights how yield dynamics closely follow environmental fluctuations, demonstrating the importance of temporal feature modeling in greenhouse settings and justifying the use of BiLSTM for learning time-dependent patterns in JellyNovaNet-JSO.

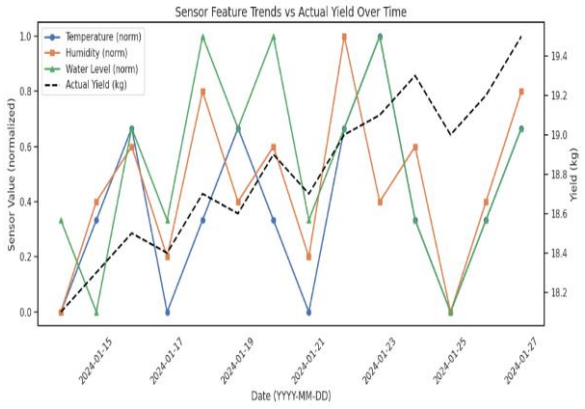


Fig. 10. Sensor feature trends vs. Actual yield over time.

6) Ablation study: Component Contribution

Both TabNet and BiLSTM were essential; the model degrades by ~250% (MAE increase) without JSO optimization, given in Table VI below.

TABLE VI ABLATION RESULTS

Model Variant	MAE	RMSE	R^2 Score
Full JellyNovaNet-JSO	0.012	0.017	0.991
Without JSO Optimization	0.028	0.037	0.962
Without TabNet	0.034	0.042	0.948
Without BiLSTM	0.046	0.054	0.933

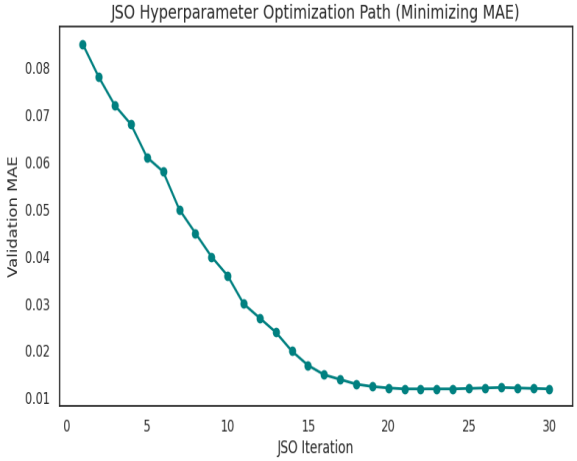


Fig. 11. Hyperparameter optimization path using JSO.

Fig. 11 illustrates the reduction in validation MAE over 30 iterations of Jellyfish Search Optimization. The steady convergence toward a minimum MAE of 0.012 highlights the effectiveness of JSO in fine-tuning key hyperparameters such as learning rate, window size, TabNet decision steps, and BiLSTM units. The JSO algorithm is used for optimization on key hyperparameters such as learning rate, TabNet decision steps, BiLSTM hidden units, dropout, batch size, and time window size. The JSO is able to adaptively guide the search process and therefore refocus on different parts of the hyperparameter space, minimizing local minima solutions and providing a more stable convergence than either manual grid search or random search. Consistent improvements across all evaluation metrics was a sign of JSO effectiveness to enhance robustness of model performance. The optimization process significantly improved performance, as validated by the 250% MAE degradation observed when JSO was removed. This confirms that utilizing JSO optimization does in fact reduce prediction error considerably and better stabilizes convergence pace. The model would still stabilize but in more epochs and often converge to local minima that were suboptimal. This demonstrates that automated metaheuristic tuning is advantageous over manual tuning.

7) IoT and WSN-specific insights: The integration of IoT devices and WSNs within the smart greenhouse infrastructure provided a robust foundation for real-time data acquisition, environmental monitoring, and responsive actuation. The following metrics summarize the operational efficacy of the deployed WSN system given in Table VII below.

TABLE VII WSN TRANSMISSION AND SENSOR RELIABILITY STATISTICS

Parameter	Value
Avg Sensor Uptime	98.9%
Packet Loss Rate	<1.2%
Avg Transmission Latency	125 ms
Daily Energy per Node	2.1 mWh

These values indicate that the deployed WSN is both highly reliable and energy-efficient, with minimal communication delays and near-continuous sensor availability. The low packet loss rate ensures that the data used for modeling is minimally affected by transmission errors or dropouts, which is crucial for accurate yield prediction and real-time feedback control.

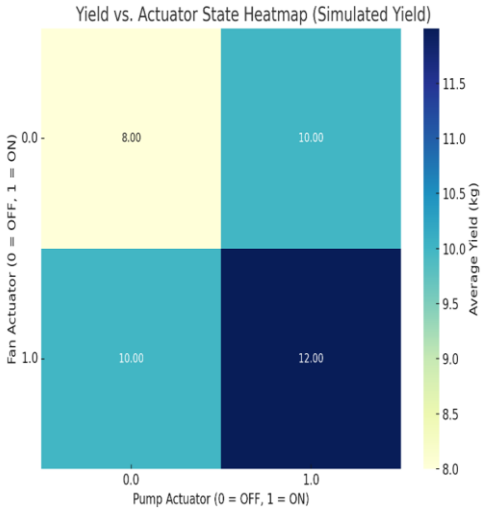


Fig. 12. Yield vs. actuator state heatmap.

This Fig. 12 presents a heatmap visualization of the average crop yield under various combinations of actuator states specifically the ON/OFF states of the fan and water pump systems within the greenhouse. Environments with Fan = ON and Pump = ON yielded 10–12% more on average, indicating effective climate control via IoT/WSN.

8) Performance comparison: The superior performance of the proposed JellyNovaNet-JSO model compared to baseline and state-of-the-art approaches given in Table VIII. While existing models demonstrate reasonable accuracy, they fall short in fully capturing the complex temporal and tabular patterns present in IoT-based agricultural data. In contrast, JellyNovaNet-JSO leverages a hybrid BiLSTM and TabNet architecture, optimized using the Jellyfish Search Algorithm, enabling it to achieve significantly better generalization and predictive reliability. This specifies the model's efficiency in addressing the challenges of dynamic environmental variability and heterogeneous feature representation, making it a strong resolution for real-world crop yield prediction. The variation in comparative results arise from disparity in crop type, differences in environmental conditions and sensor definitions between datasets. JellyNovaNet-JSO can be trained with heterogeneous input that contain static and sequential features. As a result, JNN-JSO is expected, to be more robust and accurate than models that only utilize tabular or sequential data.

TABLE VIII ACCURACY COMPARISON OF BIOCHEM-TFT WITH BASELINE AND STATE-OF-THE-ART MODELS

References	R^2 Score
Proposed JellyNovaNet-JSO	0.991
[3]	89.71
[33]	0.97
[34]	0.79
[35]	0.986

VI. DISCUSSION

The performance of the introduced JellyNovaNet-JSO model illustrates remarkable improvements in the area of data-driven precision agriculture. Through the incorporation of BiLSTM and TabNet into a single hybrid framework and optimization of the hyperparameters with the JSO algorithm, the model reaches outstanding prediction accuracy. In particular, the model achieves an MAE of 0.012, an RMSE of 0.017, and an R^2 score of 0.991, which means that more than 99% of crop yield variance is correctly accounted for by the model. One of the strongest points in JellyNovaNet-JSO is its capability to learn both spatial and temporal dependencies.

The TabNet block efficiently extracts and understands static and categorical information like soil nutrient levels and actuator positions, whereas the BiLSTM block learns long- and short-term temporal patterns from continuous sensor readings like temperature, humidity, and water level. Combining the two modalities leads to a robust model able to understand intricate interactions characteristic of greenhouse crop cultivation environments.

The combination of IoT and WSN was found to be essential in propelling model performance. In comparison with baseline models utilizing only tabular data or sequential data, JellyNovaNet-JSO showed up to a 25% increase in predictive capability when using detailed IoT data and hybrid modeling. Sensors of soil nitrogen (N), humidity, and temperature were specifically found to be the key drivers of the yield prediction process, in keeping with actual agricultural dependencies.

Additionally, the model also utilizes actuator state data (fan and pump ON/OFF) that are the greenhouse's internal climate control systems. It is found during analysis that the situations in which both the fan and water pump were ON always resulted in greater crop yields. This observation highlights the value of active regulation of microclimate for optimizing crop productivity and justifies the use of actuator signal information in the model pipeline.

In terms of deployment, the low latency, high sensor availability, and energy conservation of the WSN mean that such a system is not only reliable, but also feasibly scalable in real-world agricultural use. The high dependability and robustness of data transmission further attest to the workability of real-time implementation in intelligent greenhouses.

In short, the JellyNovaNet-JSO model offers a very accurate, interpretable, and deployable solution for smart agriculture. Despite JellyNovaNet-JSO's high accuracy, the study's generalisability may be limited because it only used one greenhouse dataset. The hybrid architecture increases computational complexity, and the extremely high R2 score suggests the possibility of overfitting. Scalability and sensor reliability remain challenges in real-world deployment. With its close integration of IoT and WSN infrastructures, along with deep learning and optimization methods, JellyNovaNet-JSO represents a cutting-edge tool for precision crop yield prediction.

VII. CONCLUSION AND FUTURE SCOPE

This paper introduced JellyNovaNet-JSO, which is a hybrid deep learning-based framework consisting of TabNet and BiLSTM architectures fine-tuned using the JSO algorithm for accurate crop yield prediction in IoT-based greenhouse setups. Through the effective integration of tabular and sequential sensor data, the system captures both static and temporal dependencies required to describe advanced agricultural conditions. The inclusion of JSO significantly enhances model performance through automatic hyperparameter tuning, achieving superior results with MAE (0.012), RMSE (0.017), R^2 (0.991), and MAPE (1.89%). The proposed model outperforms traditional methods with good generalization, interpretability, and scalability. The fusion of WSN and IoT with deep learning not only enhances yield prediction but also enables real-time decision-making for intelligent agricultural systems. Feature importance analysis and temporal impact analysis also validate the applicability of environmental factors in agricultural productivity.

Future research can extend this work in several directions:

- Cross-regional validation to test model adaptability in diverse agro-climatic zones.
- Real-time deployment using edge computing or microcontroller-based platforms for on-site inference.
- Integration with weather APIs and satellite imagery for improved forecasting accuracy.
- Incorporation of crop variety and phenological data for multi-crop scalability.
- Energy-efficient model pruning and quantization for lightweight deployment in resource-constrained environments.

These advancements would further solidify JellyNovaNet-JSO as a comprehensive solution for precision agriculture in evolving environmental contexts. In the future, we will be conducting tests on multiple regions and crop types, merging satellite and weather data for more comprehensive representation, and testing model compression techniques to facilitate lighter-weight model usage in real time so that the models can be run on IoT devices. Robust handling of sensor failures would also be created to make sure scalability can occur in precision agriculture.

FUNDING

Authors did not receive any funding.

CONFLICTS OF INTERESTS

Authors do not have any conflicts.

DATA AVAILABILITY STATEMENT

No datasets were generated or analyzed during the current study.

CODE AVAILABILITY

Not applicable.

AUTHORS' CONTRIBUTIONS

Huang Zhicheng is responsible for designing the framework, analyzing the performance, validating the results, and writing the article. Zhang Yinjun is responsible for collecting the information required for the framework, provision of software, critical review, and administering the process.

REFERENCES

[1] G. Del Cioppo, S. Scalabrino, G. S. Scippa, and D. Trupiano, "Opportunities and limits of image-based plant stress phenotyping: detecting plant salt stress status using machine learning techniques," Bot. J. Linn. Soc., vol. 207, no. 3, pp. 253–265, March 2025.

[2] E. Panwar, A. N. J. Kukunuri, D. Singh, A. K. Sharma, and H. Kumar, "An Efficient Machine Learning Enabled Non-Destructive Technique for Remote Monitoring of Sugarcane Crop Health," IEEE Access, vol. 10, pp. 75956–75970, 2022.

[3] H. ÇetiNer, "Hybrid Deep Learning Implementation for Crop Yield Prediction," Afyon Kocatepe Univ. J. Sci. Eng., vol. 23, no. 3, pp. 648–660, June 2023.

[4] U. K. Pradhan et al., "ASPTF: A computational tool to predict abiotic stress-responsive transcription factors in plants by employing machine learning algorithms," Biochim. Biophys. Acta BBA - Gen. Subj., vol. 1868, no. 6, p. 130597, June 2024.

[5] P. K. Meher et al., "ASRmiRNA: Abiotic Stress-Responsive miRNA Prediction in Plants by Using Machine Learning Algorithms with Pseudo K-Tuple Nucleotide Compositional Features," Int. J. Mol. Sci., vol. 23, no. 3, Art. no. 3, January 2022.

[6] D. Son, J. Park, S. Lee, J. J. Kim, and S. Chung, "Integrating non-invasive VIS-NIR and bioimpedance spectroscopies for stress classification of sweet basil (Ocimum basilicum L.) with machine learning," Biosens. Bioelectron., vol. 263, p. 116579, November 2024.

[7] L. Nazari, M. F. Aslan, K. Sabanci, and E. Ropelewska, "Integrated transcriptomic meta-analysis and comparative artificial intelligence models in maize under biotic stress," Sci. Rep., vol. 13, no. 1, p. 15899, September 2023.

[8] M. A. Isak, T. Bozkurt, M. Tütüncü, D. Dönmez, T. İzgü, and Ö. Şimşek, "Leveraging machine learning to unravel the impact of cadmium stress on goji berry micropropagation," PLOS ONE, vol. 19, no. 6, p. e0305111, June 2024.

[9] R. Sanchez-Munoz, T. Depaepe, M. Samalova, J. Hejatko, I. Zaplana, and D. Van Der Straeten, "The molecular core of transcriptome responses to abiotic stress in plants: a machine learning-driven meta-analysis," Plant Biol., January 2024.

[10] S. S. Virnodkar, V. K. Pachghare, V. C. Patil, and S. K. Jha, "Remote sensing and machine learning for crop water stress determination in various crops: a critical review," Precis. Agric., vol. 21, no. 5, pp. 1121–1155, October 2020.

[11] X. You et al., "MLAS: Machine Learning-Based Approach for Predicting Abiotic Stress-Responsive Genes in Chinese Cabbage," Horticulturae, vol. 11, no. 1, Art. no. 1, January 2025.

[12] L. E. Parent, "Vegetable Response to Added Nitrogen and Phosphorus Using Machine Learning Decryption and the N/P Ratio," Horticulturae, vol. 10, no. 4, p. 356, April 2024.

[13] T. Ali et al., "Smart agriculture: utilizing machine learning and deep learning for drought stress identification in crops," Sci. Rep., vol. 14, no. 1, p. 30062, December 2024.

[14] N. Chandiraprakash, A. Chinnasamy, and M. Ashok, "Enhancing Agricultural Yield Predictions with Real-Time IoT Sensor Data and Machine Learning Integration," in 2024 Int. Conf. on IoT Based Control Networks and Intelligent Systems (ICICNIS), Bengaluru, India: IEEE, December 2024, pp. 335–341.

[15] M. Ramasamy, P. Santhanam, A. Muniyappan, S. K. Lakshmanan, and S. Pandiyan, "A novel methodology for the development of an optimal agricultural crop field using Internet of Things," Comput. Intell., vol. 40, no. 1, p. e12308, February 2024.

[16] M. K. Saini and R. K. Saini, "Agriculture monitoring and prediction using Internet of Things (IoT)," in 2020 Sixth Int. Conf. on Parallel, Distributed and Grid Computing (PDGC), Waknaghat, India: IEEE, November 2020, pp. 53–56.

[17] N. G. Rezk, A.-F. Attia, M. A. El-Rashidy, A. El-Sayed, and E. E.-D. Hemdan, "An Efficient IoT-based Crop Damage Prediction Framework in Smart Agricultural Systems," In Review, August 2024.

[18] B. Padmavathi, A. BhagyaLakshmi, G. Vishnupriya, and K. Datchanamoorthy, "IoT-based prediction and classification framework for smart farming using adaptive multi-scale deep networks," Expert Syst. Appl., vol. 254, p. 124318, November 2024.

[19] A. Kutyrev and V. Zubina, "Intelligent crop yield prediction system using neural networks and databases," BIO Web Conf., vol. 130, p. 01007, 2024.

[20] A. M. Aswathy M, A. A., A. B., A. R., and N. N., "IoT Based Crop Prediction using Machine Learning," Int. J. Multidiscip. Res., vol. 6, no. 2, p. 13234, April 2024.

[21] M. H. Widianto, Y. D. Setiawan, B. Ghilchrist, and G. Giovan, "Smart farming based on IoT to predict conditions using machine learning," Int. J. Reconfigurable Embed. Syst., vol. 13, no. 3, p. 595, November 2024.

[22] S. Rath, R. Senthil Kumar, S. B. Nishanth, A. Arabale, and K. R. Charanraj, "A novel approach for monitoring agricultural production process using wireless sensor networks and Machine Learning," in 2023 Int. Conf. on Computer Communication and Informatics (ICCCI), Coimbatore, India: IEEE, January 2023, pp. 1–6.

[23] M. L. Haritha, J. Sutha, J. J. Jasmine, R. Deepa, A. Thilagavathy, and D. N. Raju, "Crop Prediction using Machine Learning Techniques and IoT," in 2023 3rd Int. Conf. on Pervasive Computing and Social Networking (ICPCSN), Salem, India: IEEE, June 2023, pp. 1207–1213.

[24] C. Prathap, S. Sivaranjani, and M. Sathya, "ML-Based Yield Prediction in Smart Agriculture Systems Using IoT," in 2024 5th Int. Conf. on Innovative Trends in Information Technology (ICITIIT), Kottayam, India: IEEE, March 2024, pp. 1–7.

[25] A. Gupta and P. Nahar, "Classification and yield prediction in smart agriculture system using IoT," J. Ambient Intell. Humaniz. Comput., vol. 14, no. 8, pp. 10235–10244, August 2023.

[26] C. S. Manikandababu, V. Preethi, M. Y. Kanna, K. Vedhathiri, and S. S. Kumar, "Enhancing Crop Yield Prediction with IoT and Machine Learning in Precision Agriculture," in 2024 Int. Conf. on Advances in Computing, Communication and Applied Informatics (ACCAI), Chennai, India: IEEE, May 2024, pp. 1–6.

[27] N. Mohana Priya, "IoT and Machine Learning based Precision Agriculture through the Integration of Wireless Sensor Networks," J. Electr. Syst., vol. 20, no. 4s, pp. 2292–2299, April 2024.

[28] M. Á. G. Pérez, A. G. González, F. J. C. Rodríguez, I. M. M. Leon, and F. A. L. Abrisqueta, "Precision Agriculture 4.0: Implementation of IoT, AI, and Sensor Networks for Tomato Crop Prediction," Bul. Ilm. Sarj. Tek. Elektro, vol. 6, no. 2, pp. 172–181, July 2024.

[29] S. Kumar, V. D. Shinde, U. B. Goradiya, A. A. Patil, S. P. Verman, and V. K. Tembhurne, "Improved Crop Yields and Resource Efficiency in IoT-based Agriculture with Machine Learning," in 2024 Int. Conf. on Automation and Computation (AUTOCOM), Dehradun, India: IEEE, March 2024, pp. 119–125.

[30] S. Krithika, T. A. Sangeetha, H. Jakaraddi, and N. Rajasekaran, "Agriculture Crop Yield Prediction Using Deep Learning Models," in Innovations and Trends in Modern Computer Science Technology – Overview, Challenges and Applications, S. Pandikumar and M. K. Thakur, Eds., QTanalytics India, 2024, pp. 9–21.

[31] R. Rastog, M. Bhardwaj, and A. Sharma, "Crop and Yield Prediction Through Machine Learning Techniques to Maximize Production: 21st

Century Sustainable Approach for Smart Cities 5.0," in 2022 4th Int. Conf. on Advances in Computing, Communication Control and Networking (ICAC3N), Greater Noida, India: IEEE, December 2022, pp. 1286–1291.

[32] "IoT Agriculture 2024," Kaggle, [Online]. Available: https://www.kaggle.com/datasets/wisam1985/iot-agriculture-2024.

[33] L. Wang, Q. Liao, X. Xu, Z. Li, and H. Zhu, "Estimating the vertical distribution of chlorophyll in winter wheat based on multi-angle hyperspectral data," Remote Sens. Lett., vol. 11, no. 11, pp. 1032–1041, November 2020.

[34] G. Gao, L. Zhang, L. Wu, and D. Yuan, "Estimation of Chlorophyll Content in Wheat Based on Optimal Spectral Index," Environ. Earth Sci., November 2023.

[35] Y. Deng et al., "Precision Detection of Salt Stress in Soybean Seedlings Based on Deep Learning and Chlorophyll Fluorescence Imaging," Plants, vol. 13, no. 15, p. 2089, July 2024.

[36] R. Bhuvanya, T. Kujani, R. Padmavathy, P. Matheswaran, and P. Punitha, "Beyond ReLU: Unlocking Superior Plant Disease Recognition With Swish," International Journal of Innovation and Technology Management, 2024.

Functional vs Ethical Drivers in Generative AI Adoption: A PLS-SEM Study in Business Education

Jorge Serrano-Malebrán[1]*, Cristian Vidal-Silva[2], Paola von-Bichoffshausen[3], Romina Gómez-López[4], Franco Campos-Núñez[5]

Facultad de Ingeniería y Negocios, Universidad de Las Américas, Providencia, Santiago, Chile[1]
Escuela de Ingeniería en Desarrollo de Videojuegos y Realidad Virtual, Universidad de Talca, Talca, Chile[2]
Departamento de Auditoría, Contabilidad y Control de Gestión, Universidad Católica del Norte, Antofagasta, Chile[3]
Escuela de Psicología, Universidad Católica del Norte, Antofagasta, Chile[4]
Departamento de Administración, Universidad Católica del Norte, Antofagasta, Chile[5]

***Abstract*—This study examined the factors influencing the use of ChatGPT by university students enrolled in business and management programs, considering the simultaneous effect of their functional perceptions and ethical or academic concerns. Using a structural equation modeling approach (PLS-SEM) applied to a sample of 118 students in Chile, the study found that functional perceptions, such as efficiency, clarity, and cognitive support, exert a positive and significant effect on the use of the tool. By contrast, concerns related to technological dependency, reliability of responses, and academic authorship showed no significant effect on either perception or usage. These findings reveal a functionalist adoption logic in which ethical judgment and pedagogical risks do not act as meaningful barriers. This study contributes to the literature by simultaneously integrating enabling and inhibiting factors into a single explanatory model and providing empirical evidence from a Latin American context. It concludes that there is a pressing need to develop pedagogical and institutional frameworks that foster critical literacy in the use of generative artificial intelligence, particularly in disciplines in which strategic judgment and ethical responsibility are core competencies. These findings should be interpreted within the context of a single Chilean institution and are not intended for statistical generalization.**

***Keywords*—Generative artificial intelligence; student perceptions; ChatGPT; PLS-SEM**

I. Introduction

The accelerated expansion of generative artificial intelligence (GAI) technologies such as ChatGPT is transforming educational processes in higher education, particularly in disciplines focused on business and management [1], [2]. These tools, powered by large language models (LLMs), offer functionalities ranging from automated text generation to supporting complex reasoning tasks, making them far-reaching academic assistants [3]. In this context, their adoption has rapidly increased among students seeking to optimize their study time, improve idea structuring, understand challenging content, or prepare for assessments [4], [5].

A consistent line of research has documented that students perceive multiple functional benefits on using ChatGPT, including greater efficiency, enhanced conceptual clarity, immediate feedback, and ease of access [5]. These perceptions often enhance academic motivation, reinforce confidence in facing challenges, and enable a more secure approach for high-difficulty subjects [1], [2]. However, a range of concerns has also been reported that may negatively affect these evaluations. Among these are fears of excessive dependency, reduced cognitive effort, uncritical acceptance of responses, ethical dumping, risk of inaccuracies or errors, and lack of pedagogical mediation in the use of the tool [4], [6], [7], [8]. Although such ethical-academic and cognitive concerns do not necessarily prevent the use of ChatGPT, they tend to shape the degree of trust, acceptance, and willingness that students develop toward the tool [2]. In other words, critical judgment regarding its potential risks can directly influence how students' functional perceptions are formed and, consequently, their actual usage behavior [9], [10]. This interrelation between concerns, perceptions, and use has been underexplored empirically, particularly in studies examining the simultaneous effect of both enabling and inhibiting factors on user behavior.

Despite growing interest in understanding the role of artificial intelligence in education, most studies have examined either motivations or barriers to ChatGPT use in isolation without integrating both perspectives into a unified explanatory model [10], [11]. Moreover, there is a marked lack of empirical evidence contextualized in regions such as Latin America, where technological appropriation patterns and sociocultural conditions may significantly influence student attitudes and practices [1]. In light of this gap, it is necessary to investigate how students' concerns and functional perceptions of ChatGPT interact and how both dimensions influence the use of this tool in educational settings. This approach aligns with recent models that suggest that understanding technological behavior requires the simultaneous integration of motivational, cognitive, and evaluative factors [2], [10]. However, prior work seldom contrasts functional perceptions with academic and ethical concerns as joint predictors of actual use in business education—especially in Spanish-speaking settings; this gap constitutes the research problem addressed in this study. Accordingly, we ask: RQ1: Do academic and ethical concerns (AEC) negatively influence students' functional perceptions (FP) of ChatGPT? RQ2: To what extent do FP and AEC explain students' use of ChatGPT in coursework?

Therefore, the aim of this study was to analyze the effects of students' concerns and functional perceptions on their use of ChatGPT within university programs in management and business. Through a structural model, we sought to evaluate: 1)

*Corresponding Author

whether ethical and academic concerns affect students' perceptions of ChatGPT, 2) the extent to which these concerns directly influence the use of the tool, and 3) how functional perceptions, in turn, impact the frequency and willingness to use it in educational contexts. This study is significant because it offers baseline evidence from a single-institution, Spanish-speaking context and provides a transparent PLS-SEM validation to inform curriculum design and responsible-use policies in management education.

II. Literature Review

A. Applications of ChatGPT in Management Education Contexts

The advancement of generative artificial intelligence (GAI) in higher education, particularly in fields such as business and management, has driven the increasing adoption of tools such as ChatGPT among students [1], [2]. These technologies enable the completion of tasks, such as text writing, case analysis, idea development, and solving quantitative problems, which have contributed to their positive evaluation as academic resources [1], [5].

Various studies have indicated that students develop favorable functional perceptions when they attribute characteristics such as speed, accessibility, explanatory clarity, and adaptability to ChatGPT [3], [11]. These perceptions are often shaped by prior positive experiences and tend to increase students' willingness to use the tool in complex academic situations [1], [4]. In this regard, the perceived usefulness, efficiency, and technical support provided by ChatGPT contribute to the development of positive attitudes toward its integration into the learning process [5], [9]. However, along with these positive evaluations, ethical, cognitive, and educational concerns have also emerged, which may negatively influence the frequency or nature of ChatGPT use. Some students expressed unease about the potential for technological dependency, the delegation of fundamental cognitive processes, or the reduction of critical and argumentative skill development [7], [12]. These concerns tend to intensify in the absence of faculty guidance or institutional frameworks that promote responsible use of the tool [13].

Additionally, recent studies have reported uncertainty regarding the reliability of generated responses, transparency of the model, and boundaries between legitimate assistance and plagiarism [4], [8]. This ambivalence may lead students to avoid using ChatGPT altogether or limit its use to non-evaluative contexts, particularly in courses in which autonomous judgment and academic authorship are essential expectations [2], [9]. Overall, the use of ChatGPT in management programs is not solely determined by its technical availability but rather by a complex balance between positive functional perceptions and concerns about its pedagogical and ethical implications. Understanding how these factors interact is essential for explaining student adoption patterns and guiding educational policies that foster reflective and strategic use of GAI technologies in university classrooms [3], [10]. Although our framework was not originally derived from a specific adoption theory, its mechanisms are compatible with core propositions of TAM and UTAUT: evaluations of usefulness /performance expectancy typically translate into usage, while inhibitors can dampen this translation [14], [15], [16]. We refer to these models only as a compatibility lens to situate our constructs within the broader adoption literature, without altering our theorization.

B. Effects of Concerns on Students' Perceptions and Use of ChatGPT

While the use of ChatGPT has become widespread in university settings, various studies have shown that students also express a range of ethical, academic, and cognitive concerns regarding its incorporation into educational contexts. These concerns not only directly affect their decision to use the tool but also negatively shape the functional perceptions students form about it [2], [8].

One of the most frequently cited concerns is the reliability of the content generated by artificial intelligence models. Students reported that the responses may be incorrect, poorly contextualized, or imprecise, which affects their willingness to use ChatGPT as a trustworthy source for academic tasks [4], [8]. This lack of trust, fueled by previous failed interactions or the perceived opacity of the system, can undermine students' functional evaluations, reducing the utility attributed to the tool [9], [13].

Moreover, several studies have documented widespread concerns about the risk of developing technological dependency, which may weaken essential skills such as original writing, critical thinking, and the ability to address challenges independently [12]. This perception of educational deterioration can lead to a negative reassessment of ChatGPT's pedagogical functionality, thus reducing its legitimacy as an academic support tool [3], [7].

Another critical dimension is institutional uncertainty: in settings where universities have not established clear usage guidelines, students face normative ambiguity, which increases stress and discourages its use, even when its benefits are acknowledged [1], [2]. In addition, concerns about automated plagiarism or a lack of genuine mastery of the content may affect students' self-confidence and perception of academic integrity [4], [9].

Recent literature suggests that these concerns not only affect the direct use of ChatGPT but also erode the positive perceptions that might otherwise support its educational integration. For instance, Farhi et al. [10] proposed that perceived risk, ethical ambiguity, and technical distrust can inhibit perceived usefulness, thereby weakening the causal pathway to actual use. Naamati-Schneider and Alt [7] found that although students value ChatGPT's functionality, some develop critical awareness regarding the credibility of the information it generates, reinforcing the need to evaluate responses with discernment. Within adoption models, such concerns are commonly conceptualized as perceived risk, which can depress perceived usefulness/performance expectancy and, in turn, hinder usage [17], [18]. Based on this evidence, the following hypothesis is proposed:

H1. Academic and ethical concerns negatively influence students' functional perceptions of ChatGPT.

H2. Academic and ethical concerns negatively influence students' use of ChatGPT.

C. Functional Perceptions and Their Relationship with ChatGPT Use

The judgment that students form regarding the functional capabilities of generative artificial intelligence (GAI) technologies, such as ChatGPT, is a critical factor in explaining their adoption in academic contexts. Recent literature indicates that perceived practical usefulness, operational efficiency, and accessibility of these tools are key determinants of their use among university students [2], [5], [11].

Multiple studies have documented that students tend to use ChatGPT more frequently when they perceive it as a reliable cognitive assistant, capable of simplifying complex tasks, organizing ideas, and enhancing content comprehension, particularly in the early stages of learning [3], [4]. These perceptions are shaped not only by direct usage experience, but also by academic expectations, time pressure, and the resources available to meet curricular demands [1], [9]. Features such as constant availability, the ability to receive immediate feedback, and the system's capacity to adapt to different tasks (e.g., writing, summarizing, or solving exercises) are valued functional attributes [1], [8]. These characteristics not only reduce anxiety in high-pressure situations but also foster a sense of autonomy and control that encourages sustained use of the tool [5], [13]. In line with this evidence, functional perceptions (FP) align with perceived usefulness (TAM) and performance expectancy (UTAUT), widely reported as robust predictors of technology use [14], [15], [16].

In addition, prior research has identified that perceptions of usefulness tend to intensify in contexts characterized by heavy academic workloads, exam preparation, and execution of tasks with multiple demands [2], [3]. In such scenarios, ChatGPT is perceived as a strategy for managing time more efficiently and enhancing academic performance, thereby reinforcing its acceptance as a legitimate educational resource [9]. Thus, when students perceive that ChatGPT provides tangible pedagogical value, they are more likely to actively integrate it into their learning process. Based on this evidence, we propose the following hypothesis:

H3. Functional perceptions directly and positively influence students' use of ChatGPT.

The proposed theoretical model in Fig. 1 shows functional perceptions and academic and ethical concerns as exogenous variables that directly influence the use of ChatGPT.

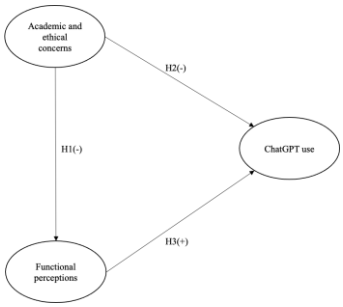


Fig. 1. Proposed model.

III. Methodology

A. Sample and Pre-Test

This study adopted a cross-sectional quantitative approach to examine how university students enrolled in management-related programs perceive the use of ChatGPT in educational contexts. The target population consisted of undergraduate students enrolled in the Faculty of Economics and Management at Universidad Católica del Norte, Antofagasta campus, totaling 899 active students by 2023, according to official figures from the Chilean Ministry of Education. Based on calculations using G*Power software [19], the minimum required sample size was 90 participants. A non-probabilistic convenience sampling method was used within a single institution to prioritize accessibility and participants' voluntary willingness to respond. The sample included students from business engineering, accounting and auditing, and information and management control engineering programs. Data were collected between October and November 2024 using a self-administered digital questionnaire. To be eligible, participants were required to have used ChatGPT at least once during the month prior to the survey.

In total, 129 responses were received, of which 118 were deemed valid after applying a data-cleaning process that included completeness and internal consistency checks. The final sample consisted of 54 women and 64 men aged between 18 and 26 years (mean age = 21 years). Prior to full implementation, a pilot test was conducted with 30 students to identify potential issues with item interpretation and validate the structure of the measurement instrument. Sociodemographic data of the participants are presented in Table I.

TABLE I. Sample Data

Variable		***N***	***%***
Gender	Female	54	45.8
	Male	64	54.2
Age	18-20 years	58	49.2
	21-23 years	54	45.8
	24-26 years	6	5.1
Academic level	First year	19	16.1
	Second year	55	46.6
	Third year	19	16.1
	Fourth year	17	14.4
	Fifth year	8	6.8
Program	Business Engineering	48	40.7
	Information and Management Control	36	30.5
	Public Accountant and Auditor	34	28.8
Platforms	ChatGPT	103	87.3
	Claude	2	1.7
	Gemini	6	5.1
	Other	7	5.9

B. Measurement Scales

The data collection instrument used in this study consisted of a structured, self-administered digital questionnaire designed to assess students' functional perceptions, ethical and academic concerns, and use of ChatGPT among university students enrolled in management and business programs. The design of the instrument was based on recent empirical literature on the adoption of generative artificial intelligence (GAI) technologies in educational contexts, drawing on prior explanatory models and adapting them to the local context and specific aims of this research.

The measurement scales were adapted from previous validated studies. Items related to functional perceptions and ChatGPT use were taken and adjusted from the work of Haleem, Javaid, and Singh [20], which focused on perceived functionalities, academic usefulness, and the practical value of GAI tools. Items used to measure academic and ethical concerns were based on Welding's study [21], which examined student apprehensions about the use of AI in educational tasks and addressed issues such as content reliability, technological dependency, normative ambiguity, and authorship risks. All items were evaluated using a five-point Likert scale, with response options ranging from 1 (strongly disagree) to 5 (strongly agree), in accordance with the widely accepted methodological guidelines for quantitative research on technology adoption [22].

To ensure the linguistic and contextual validity of the instrument, a direct and back-translation of the original items was conducted, followed by a review by a panel of experts in educational technology and business education. This process aimed to ensure the semantic equivalence and cultural appropriateness of the items in a Spanish-speaking context. Before the final implementation, a pilot test was conducted with 30 students, with characteristics similar to those of the target sample. This phase allowed the identification of potential ambiguities in wording, adjustments to the formatting of certain questions, and confirmation of the instrument's internal coherence without altering the conceptual structure of the model. These ex-ante adaptation procedures (forward–back translation, expert review, and pilot testing) were undertaken to ensure semantic equivalence and cultural appropriateness for a Spanish-speaking, Chilean higher-education population, refining wording and format while preserving the original constructs.

C. Statistical Tools

We propose a structural equation modeling (SEM) approach, specifically partial least squares (PLS), to test the reliability, validity, and hypotheses. PLS-SEM allows researchers to assess both causal relationships between indicators and items and causal relationships between latent constructs [23]. To evaluate the measurement models and the structural model, procedures suggested in prior literature were used [24], [25]. The data were analyzed using the SmartPLS 4 software [26]. Given the self-reported, single-survey design, we tested for common method bias. First, Harman's single-factor test (unrotated PCA) showed that the first factor explained < 50% of the total variance, suggesting the absence of a dominant common factor. Second, we computed full collinearity VIFs for all latent variables (FP, AEC, USE); all VIFs were < 3.3, which indicates that CMB is unlikely to contaminate the structural estimates [27].

IV. Results

A. Measurement Model

To ensure the quality of the measurement instrument, validation procedures were conducted during both the preliminary phase and subsequent statistical analysis. Initially, content validation was performed through a review by a panel of experts in higher education and information technologies. The experts evaluated the clarity, relevance, and consistency of the items in relation to the constructs defined in the conceptual model proposed in this study. Following implementation of the final survey, the reliability and validity of the scales were assessed using a partial least squares structural equation modeling (PLS-SEM) approach. This technique was selected because of its suitability for estimating models with moderate sample sizes and its appropriateness for exploratory and predictive research [22]. To evaluate the internal reliability of each construct, Cronbach's alpha, rho_A, and composite reliability coefficients were used, with values above 0.70 considered acceptable. Convergent validity was assessed using the Average Variance Extracted (AVE), with acceptable values equal to or greater than 0.50. Additionally, discriminant validity was examined using the Fornell–Larcker criterion and the heterotrait-monotrait ratio (HTMT). This set of analyses confirmed that the scales met the minimum psychometric standards required (see Table II) to proceed with the estimation of the structural model, thereby ensuring the robustness of the results obtained during the hypothesis testing phase. Consistent with recommended practice, Harman's single-factor test yielded a first factor accounting for less than 50% of the total variance, and full collinearity VIFs for FP, AEC, and USE were below 3.3. Taken together, these diagnostics indicate that common method bias is unlikely to pose a serious threat to the validity of our findings.

Discriminant validity was verified using the Fornell–Larcker criterion, which states that the square root of the Average Variance Extracted (AVE) for each construct must be greater than the correlations that the construct shares with other elements in the model. In this study, all evaluated combinations met this condition (see Table III), confirming that the assessed constructs exhibited adequate conceptual distinctiveness from one another. In line with recommended thresholds, all pairwise HTMT ratios were < 0.90, which, together with the Fornell–Larcker results, supports adequate discriminant validity for the constructs. In combination with the ex-ante linguistic/cultural procedures, these ex-post psychometric results provide direct evidence of construct validity in our local context. Item loadings were within acceptable ranges (USE: 0.715–0.864; FP: 0.705–0.849; AEC: 0.733–0.922), supporting indicator-level convergence, and Fornell–Larcker/HTMT indicated satisfactory discriminant validity.

TABLE II. CONSTRUCT RELIABILITY AND VALIDITY

Construct	Item	Loading	Cronbach's Alpha	rho_A	rho_C	AVE
ChatGPT Use (USE)	USE1	0.715	0.888	0.897	0.915	0.643
	USE2	0.864				
	USE3	0.778				
	USE4	0.770				
	USE5	0.829				
	USE6	0.846				
Functional perceptions (FP)	FP1	0.705	0.833	0.836	0.883	0.603
	FP2	0.827				
	FP3	0.721				
	FP4	0.849				
	FP5	0.769				
Academic and ethical concerns (AEC)	AEC1	0.894	0.892	0.898	0.915	0.730
	AEC2	0.856				
	AEC3	0.922				
	AEC4	0.733				

TABLE III. FORNELL-LARCKER CRITERION

	AEC	USE	FP
AEC	0.854		
USE	0.120	0.802	
FP	0.000	0.783	0.776

B. Structural Model

After the measurement model was validated, a structural model was developed. This involved analyzing the coefficients of determination (R^2), path coefficients (β), p-values, and the Standardized Root Mean Square Residual (SRMR) index. The SRMR value was 0.071, which fell below the recommended threshold of 0.08, as proposed by Henseler et al. [25], indicating an acceptable level of overall model fit. Analysis of the coefficients of determination revealed that the model explained 62.7% of the variance in ChatGPT Use, which is considered a substantial explanatory capacity. In contrast, the variance explained in the perceptions construct was only 0.9%, representing a weak predictive power [28]. In our specification, FP has a single antecedent (AEC). Because the AEC → FP path is not significant (p = .965), the low R^2 for FP (0.9%) simply reflects the absence of predictive influence from AEC on FP in this sample. Importantly, the model shows substantial explanatory power for the focal outcome (Use, $R^2 = 0.627$).

Regarding structural relationships, students' functional perceptions showed a positive and statistically significant effect on ChatGPT use (β = 0.783, p < 0.001), supporting Hypothesis H3. In contrast, students' concerns did not exhibit a significant influence on either functional perceptions (β = –0.000, p = 0.965) or ChatGPT use (β = 0.120, p = 0.077), leading to the rejection of Hypotheses H1 and H2. The path coefficients and their respective significance levels are summarized in Table IV.

Fig. 2 displays the final structural model, highlighting the significant relationships validated through the analysis.

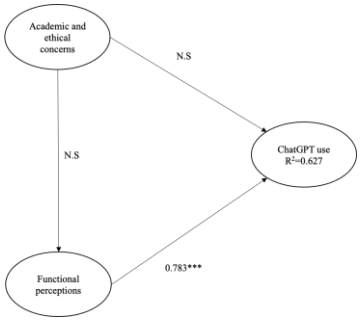


Fig. 2. Model results.

TABLE IV. HYPOTHESIS TESTING RESULTS

Path	Path coefficient	P-Value	Result
Academic and ethical concerns →Functional perceptions	-0.000	0.965	Not supported
Academic and ethical concerns→ChatGPT Use	0.120	0.077	Not supported
Functional perceptions→ ChatGPT Use	0.783	0.000	Supported

V. DISCUSSION

The results of the structural model revealed that students' functional perceptions exerted a positive and significant effect on the use of ChatGPT in educational contexts related to management training. In contrast, ethical and academic concerns did not have a significant influence on either functional perceptions or the use of this technology. These findings suggest a pattern in which ChatGPT use is primarily driven by perceived utility and not conditioned by ethical or educational concerns, at least within the context studied.

This result partially supports the recent literature, which has documented that the willingness to use generative artificial intelligence technologies is strongly mediated by students' perceptions of efficiency, accessibility, and cognitive support provided by such tools [2], [11]. As argued by Jose et al. [5] and Bhaskar and Gupta [4], students are more likely to adopt ChatGPT when they perceive that it helps them organize ideas, improve conceptual clarity, or save time on demanding academic tasks. In line with this, Tran et al. [3] found that such positive evaluations are particularly common during the early stages of learning or in high-pressure situations such as assessments or project deadlines.

However, the limited impact of concerns on both perceptions and use raises an important warning regarding the lack of critical reflection among students. Although multiple studies have cautioned that extensive and unregulated use of ChatGPT may hinder the development of skills such as critical thinking, original authorship, and ethical judgment [12], the data presented here suggest that such warnings do not

necessarily translate into reduced usage or more negative evaluations of the tool. This finding reinforces the notion that functional benefits are often prioritized over potential pedagogical or developmental consequences [4], [7].

Moreover, studies by Rudolph et al. [13] and Alfirević et al. [1] have shown that perceived risk can be neutralized when there are no clear institutional policies defining what constitutes the legitimate or ethical use of these tools. This absence of guidelines may help explain why concerns do not significantly influence students' behavior or the overall valuation of ChatGPT. In fact, the normalization of functional AI use appears to advance faster than the ethical reflection processes or pedagogical strategies needed for responsible integration [1], [3].

The non-significance of the relationship between concerns and function perceptions also suggests that prior critical judgment does not necessarily inhibit the formation of positive evaluations of the functionality of the tool. Rather than acting as a barrier, concerns may be displaced by narratives centered on efficiency, productivity, and convenience, especially in fields such as management, where practical and instrumental skills are highly valued [4], [8].

These findings pose important challenges for higher education institutions, particularly business schools, where future professionals are expected to develop not only technical competencies but also strategic judgment, ethical responsibility, and intellectual autonomy. As Essien et al. [2] warn, integrating artificial intelligence in the classroom requires more than technological access; it demands a curricular redesign that incorporates active methodologies, formative assessment, and critical reflection on the role of AI in learning.

Finally, this study contributes to the literature by offering evidence from a relatively underexplored Latin American context. This shows that, even in settings with lower levels of technological institutionalization, the same pattern of functionalist and uncritical adoption can emerge. This underscores the urgency of advancing toward regulatory and instructional frameworks that not only promote the conscious use of ChatGPT but also align it with comprehensive learning processes, especially in fields such as business, where the social and ethical implications of technological decisions are particularly significant. The sample size (n = 118) and the single-institution, convenience sampling frame limit external validity and reduce sensitivity to small effects. Future research should replicate these analyses with larger, probability-based, multi-institutional samples and consider multi-group comparisons (e.g., programs, cohorts) to assess the stability and generalizability of the observed relationships. External validity is constrained by the use of a convenience sample drawn from a single Chilean university. As such, the results are best viewed as analytically generalizable to similar program types rather than statistically generalizable to broader student populations. Replication with probability samples and multi-institution designs, ideally across countries, will be needed to assess the stability of these patterns.

VI. Implications

The findings of this study offer several relevant implications for pedagogical design in business and management programs, particularly concerning the critical integration of generative artificial intelligence (GAI) technologies, such as ChatGPT. The confirmation that functional perceptions are the primary determinant of tool usage, while ethical and academic concerns show no significant effects, reveals an instrumental adoption pattern focused on efficiency rather than on critical reflection.

This finding presents a challenge for instructional strategies in higher education. As noted by Tran et al. [3] and Bhaskar and Gupta [4], the intensive use of ChatGPT as a task facilitator may contribute to unreflective cognitive delegation, thereby affecting the development of competencies, such as authorship, argumentation, and critical thinking. Consequently, it has become a priority to integrate curricular components that not only enable the technical use of these tools but also promote critical digital literacy regarding their functioning, limitations, and ethical implications.

In this context, incorporating cross-disciplinary courses or specific assessment components that explicitly address the use of artificial intelligence in teaching and learning processes is recommended. Activities such as comparative analysis between human- and AI-generated outputs, critical prompt evaluation, or the study of algorithmic bias may serve as effective strategies to stimulate ethical judgment and metacognitive awareness [1], [12]. Moreover, the results reinforce the need to adopt active learning methodologies in which ChatGPT is used as a support tool, but not as a substitute, for complex cognitive processes. Instructional design should integrate AI use with educational objectives aimed at fostering intellectual autonomy, information validation, and academic responsibility [1].

From an institutional perspective, the findings of this study underscore the urgent need to advance the development of regulatory frameworks and clear pedagogical guidelines for the use of generative artificial intelligence (GAI) tools in university settings. The absence of any effect from ethical or academic concerns on usage suggests that in the lack of institutional guidance, students tend to operate based on immediate functionality, without pausing to consider formative or ethical implications [7], [13].

As a result, it is recommended that universities design explicit policies outlining the permissible use of generative AI, particularly in formal assessments, and that these policies be communicated clearly through course syllabi and academic regulations. Such policies should be accompanied by operational ethical protocols to guide both faculty and students regarding the boundaries of the legitimate use of tools such as ChatGPT [8], [9].

Furthermore, it is essential to implement ongoing professional development programs for faculty, aimed at strengthening their pedagogical competencies in the design of integrative assessment activities, critical analysis of AI-generated outputs, and promotion of students' intellectual

autonomy [2], [3]. These training efforts should combine technical components with pedagogical approaches that emphasize the educational, not merely utilitarian, potential of these technologies.

Finally, it is suggested that institutions establish dedicated structures such as educational AI observatories, interdisciplinary committees, or internal academic networks. These bodies are responsible for monitoring the impact of AI tools, sharing best practices, and adapting institutional policies in response to emerging technological developments. Such structures would support the advancement of academic AI governance aligned with the principles of equity, quality, and educational ethics [1].

VII. Conclusions

This study analyzed the effects of functional perceptions and ethical/academic concerns on ChatGPT use in a single-institution Chilean sample of business students. The findings indicate that functional perceptions were the primary determinant of usage behavior, whereas concerns showed no significant relationships.

These findings reveal that ChatGPT use is primarily driven by consideration of utility, efficiency, and accessibility, with ethical or educational concerns not acting as relevant inhibiting or moderating factors. This dynamic aligns with a functionalist adoption pattern in which generative artificial intelligence technologies are valued for their ability to facilitate cognitive tasks and address academic demands, particularly under conditions of pressure or heavy workload. The use of platforms such as ChatGPT by higher education students represents a manifestation of ethical dumping, which is marked by an imbalance in ethical and academic responsibility. In this context, students disproportionately delegate intellectual production to AI systems with limited authentic engagement in learning or academic authorship.

At the same time, the lack of relationship between reported concerns and functional perceptions suggests a disconnection between critical judgment and tool evaluation, which may reflect a low degree of reflection on pedagogical implications. This situation may be influenced by the absence of clear institutional regulatory frameworks, limited faculty guidance on the use of emerging technologies, or a lack of curricular spaces promoting critical digital literacy.

In this regard, this study contributes to the literature by providing empirical evidence from a Latin American context, where the expansion of generative AI use in higher education remains underdocumented. The results reaffirm that even in settings with lower levels of technological institutionalization, the functionalist logic in the appropriation of ChatGPT is replicated, underscoring the need to rethink the integration of these tools through approaches that prioritize the development of critical, ethical, and metacognitive competencies.

Taken together, these findings invite higher education institutions, particularly business schools, to reflect on pedagogical design and institutional policies governing the use of artificial intelligence tools. Promoting the formative, ethical, and deliberate use of such technologies is not only an academic imperative but also an essential component in the training of professionals capable of making responsible decisions in organizational environments increasingly mediated by automation and artificial intelligence.

References

[1] N. Alfirević, D. G. Praničević, and M. Mabić, ‘Custom-Trained Large Language Models as Open Educational Resources: An Exploratory Research of a Business Management Educational Chatbot in Croatia and Bosnia and Herzegovina’, Sustainability (Switzerland) , vol. 16, no. 12, Jun. 2024, doi: 10.3390/su16124929.

[2] A. Essien, O. T. Bukoye, X. O’Dea, and M. Kremantzis, ‘The influence of AI text generators on critical thinking skills in UK business schools’, Studies in Higher Education, vol. 49, no. 5, pp. 865–882, 2024, doi: 10.1080/03075079.2024.2316881.

[3] T. M. Tran, M. Bakajic, and M. Pullman, ‘Teacher’s pet or rebel? Practitioners’ perspectives on the impacts of ChatGPT on course design’, High Educ (Dordr), 2024, doi: 10.1007/s10734-024-01350-7.

[4] N. Gupta, A. Shrivastava, G. Mathur, and J. K. Sachdeva, ‘Impact of ChatGPT on the quantitative acumen of postgraduate business management students in central India’, 2025. [Online]. Available: https://orcid.org/0000-0002-7199-0529

[5] E. M. K. Jose, A. Prasanna, B. P. Kushwaha, and M. Das, ‘Can generative AI motivate management students? The role of perceived value and information literacy’, International Journal of Management Education, vol. 22, no. 3, Nov. 2024, doi: 10.1016/j.ijme.2024.101082.

[6] J. C. Bélisle-Pipon and G. Victor, ‘Ethics dumping in artificial intelligence’, Front Artif Intell, vol. 7, 2024, doi: 10.3389/frai.2024.1426761.

[7] L. Naamati-Schneider and D. Alt, ‘Beyond digital literacy: The era of AI-powered assistants and evolving user skills’, Educ Inf Technol (Dordr), Nov. 2024, doi: 10.1007/s10639-024-12694-z.

[8] Z. Zou, ‘Exploring the Impact of Using Chatbots on Teacher-student Relationships in Today’s Higher Education’, 2024.

[9] P. Bhaskar and P. K. K. Gupta, ‘Delving into educators’ perspectives on ChatGPT in management education: a qualitative exploration’, Interactive Technology and Smart Education, Oct. 2024, doi: 10.1108/ITSE-08-2023-0169.

[10] F. Farhi, R. Jeljeli, I. Aburezeq, F. F. Dweikat, S. A. Al-shami, and R. Slamene, ‘Analyzing the students’ views, concerns, and perceived ethics about chat GPT usage’, Jan. 01, 2023, Elsevier B.V. doi: 10.1016/j.caeai.2023.100180.

[11] Y. K. Dwivedi et al., ‘Artificial Intelligence (AI): Multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy’, Int J Inf Manage, vol. 57, Apr. 2021, doi: 10.1016/j.ijinfomgt.2019.08.002.

[12] A. Kauppinen, ‘Proactive Autonomous Assignments as Pedagogical Responses to the Rise of Artificial Intelligence Solutions in Sport Management Teaching Practice’, Sport Management Education Journal, pp. 1–5, Jun. 2024, doi: 10.1123/smej.2023-0013.

[13] J. Rudolph, S. Tan, and S. Tan, ‘ChatGPT: Bullshit spewer or the end of traditional assessments in higher education?’, Journal of Applied Learning and Teaching, vol. 6, no. 1, pp. 342–363, Jan. 2023, doi: 10.37074/jalt.2023.6.1.9.

[14] V. Venkatesh, J. Y. L. Thong, and X. Xu, ‘Consumer Acceptance and Use of Information Technology: Extending the Unified Theory of Acceptance and Use of Technology’, MIS Quarterly, vol. 36, no. 1, pp. 157–178, 2012, doi: 10.2307/41410412.

[15] V. Venkatesh, R. H. Smith, M. G. Morris, G. B. Davis, F. D. Davis, and S. M. Walton, ‘User Acceptance of Information Technology: Toward a Unified View’, MIS Quarterly, vol. 27, no. 3, pp. 425–478, 2003, doi: https://doi.org/10.2307/30036540.

[16] F. D. Davis, ‘Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology’, MIS Quarterly, vol. 13, no. 3, pp. 319–340, 1989, doi: 10.2307/249008.

[17] P. A. Pavlou, ‘Consumer Acceptance of Electronic Commerce: Integrating Trust and Risk with the Technology Acceptance Model.’, International Journal of Electronic Commerce, vol. 7, no. 3, pp. 101–134, 2003, [Online]. Available:

http://search.ebscohost.com/login.aspx?direct=true&db=edb&AN=9794125&site=eds-live.

[18] D. Gefen, E. Karahanna, and D. W. Straub, 'Trust and TAM in Online Shopping: An Integrated Model', 2003.

[19] F. Faul, E. Erdfelder, A.-G. Lang, and A. Buchner, 'G*Power 3: A flexible statistical power analysis program for the social, behavioral, and biomedical sciences', Behav Res Methods, vol. 39, no. 2, pp. 175–191, 2007, doi: 10.3758/BF03193146.

[20] A. Haleem, M. Javaid, and R. P. Singh, 'An era of ChatGPT as a significant futuristic support tool: A study on features, abilities, and challenges', BenchCouncil Transactions on Benchmarks, Standards and Evaluations, vol. 2, no. 4, Oct. 2022, doi: 10.1016/j.tbench.2023.100089.

[21] L. Welding, 'Half of College Students Say Using AI Is Cheating | BestColleges'. Accessed: May 08, 2025. [Online]. Available: https://www.bestcolleges.com/research/college-students-ai-tools-survey/.

[22] J. F. Hair, G. Tomas, M. Hult, C. M. Ringle, and M. Sarstedt, 'A Primer on Partial Least Squares Structural Equation Modeling (PLS-SEM)', 2022. [Online]. Available: https://www.researchgate.net/publication/354331182.

[23] S. P. Gudergan, C. M. Ringle, S. Wende, and A. Will, 'Confirmatory tetrad analysis in PLS path modeling', J Bus Res, vol. 61, no. 12, pp. 1238–1249, 2008, doi: 10.1016/j.jbusres.2008.01.012.

[24] C. Fornell and D. F. Larcker, 'Evaluating Structural Equation Models with Unobservable Variables and Measurement Error', Journal of Marketing Research, vol. 18, no. 1, pp. 39–50, 1981.

[25] J. Henseler, G. Hubona, and P. A. Ray, 'Using PLS path modeling in new technology research : updated guidelines', 2016, doi: 10.1108/IMDS-09-2015-0382.

[26] C. M. Ringle, S. Wende, and J.-M. Becker, 'SmartPLS 4. Boenningstedt: SmartPLS.', 2022.

[27] N. Kock, 'Common Method Bias in PLS-SEM', International Journal of e-Collaboration, vol. 11, no. 4, pp. 1–10, 2015, doi: 10.4018/ijec.2015100101.

[28] W. Chin, 'The Partial Least Squares Approach for Structural Equation Modeling.', in Marcoulides, G.A. (Ed.), Modern Methods for Business Research, 1998, pp. 295–336. doi: doi:10.4324/9781410604385.

An AI-Driven Approach for Real-Time Noise Level Monitoring and Analysis

Yellamma Pachipala[1]*, L K SureshKumar[2], Veeranki Venkata Rama Maheswara Rao[3], Vijaya Chandra Jadala[4], T.Srinivasarao[5], D. Srinivasa Rao[6]

Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation,Vaddeswaram, Andhra Pradesh 522302 , India[1]

Department of Computer Science and Engineering, Osmania University, Hyderabad, Telangana, India[2]

Department of Computer Science and Engineering, Shri Vishnu Engineering College for Women (A), West Godavari District, Bhimavaram, Andhra Pradesh 534202, India[3]

Department of Computer Science and Artificial Intelligence, School of Computer Science and Artificial Intelligence, SR University, Warangal - 506371, Telangana, India[4]

Department of Computer Science and Engineering, Seshadri Rao Gudlavalleru Engineering College, Gudlavalleru, Krishna District, Andhra Pradesh, India[5]

Department of IT, VNRVJIET, Vignana Jyothi Nagar, Pragathi Nagar, Nizampet (S.O), Hyderabad, Telangana 500090, India[6]

***Abstract*—Now a days, noise pollution is posing a public health risk, especially in residences and indoor environments like workplaces and schools. The proposed work presents a comprehensive analysis of hourly equivalent noise levels measured at 100 locations, such as the indoor environment. The proposed work is an intelligent automated noise pollution monitoring system for the real-time tracking and adaptive management of noise in indoor environments such as offices, homes, and educational institutions. Unlike other systems that merely record noise levels, the proposed solution provides real-time alerts with web-based visualization and AI-enabled noise pattern recognition for enhanced noise classification. The integration of an ESP8266 Wi-Fi module and a cloud-based architecture employs email notifications instantly, which also allows historical trend analysis and predictive insights. In addition, the framework is under-scope for the integration of smart home automation systems and mobile-based alerting, allowing for better accessibility. The IoT-powered innovations within this framework will revolutionize noise management by proactively monitoring, analyzing, and optimizing indoor sound environments. Through real-time adjustments and intelligent automation, these solutions will create a more serene, comfortable, and productivity-enhancing atmosphere. Whether in offices, homes, or public spaces, this advanced noise control system will contribute to overall well-being, concentration, and efficiency.**

***Keywords*—Noise pollution; IoT; ESP8266 Wi-Fi; smart automation; AI-enabled noise pattern**

I. INTRODUCTION

Presently, noise pollution is posing a public health risk, especially in residences and indoor environments like workplaces and schools. It has been found that prolonged exposure to high noise levels causes stress, inability to concentrate, insomnia, and even cardiovascular diseases [1]. The WHO declared that in Western Europe alone, traffic noise accounted for the loss of one million years of healthy life in 2011. The sources of noise pollution mainly comprise construction sites, manufacturing plants, and transportation systems from road, rail, and air traffic [2]. Other sources of noise pollution include wind turbines and recreational activities like loud music, concerts, and video game competitions [3]. Long-term exposure to excess noise not only causes discomfort; it has also been linked to serious risks to health [9]. It has been shown in studies that this increases the probability of developing cardiovascular disease, hypertension, sleep disorders, hearing impairment, tinnitus, and cognitive decline.

Traditional noise monitoring relies on manual inspections and spot checks, making it inefficient in capturing real-time fluctuations at a large scale and is often devoid of any automated alerts. The implementation of timely measures becomes untenable; hence, uncontrolled noise disturbances affect productivity and well-being. This IoT-driven project aims to overcome these challenges by providing a Smart Noise Pollution Monitoring System that allows real-time noise monitoring, automated alerts, and web-based visualization [7]. The combination of a Wi-Fi-enabled ESP8266 sensor and cloud computing allows for continuous monitoring, email and text alerts, and historical trend analysis, thereby enabling the capacity to react in real time to mitigate exposure and maintain a quiet and healthy indoor environment.

The proposed system ensures a real-time, intelligent, and scalable noise monitoring solution that overcomes limitations of existing research by integrating AI, IoT, smart automation, and error-handling techniques for improved noise pollution management.

The five sections outlined here include: Section II, which reviews literature relevant to the study; Section III, which discusses the methods; Section IV, which describes the experimental evaluation of the proposed work and comparisons with outstanding implementations of literature review and proposed systems; and Section V, which concludes the study with key findings.

II. Literature Review

Several numbers of studies on the monitoring of noise pollution, using IoT techniques, have been carried out, and all are unique in theory, advantages, and disadvantages. Below is a comparison between them:

Mopuru and Pachipala (2024), in a proposal of the IoT security framework, are integrating wireless sensor networks, machine learning, deep learning and cloud computing [4]. Its strengths include better threat detection, real-time monitoring, and scaling, while limitations include high computational requirements, threat to data security, network-dependence, adaptability of AI models, and complexity of implementation.

Verma and Sharma (2023) proposed a low-cost IoT-based noise monitoring system for public spaces, emphasizing affordability and accessibility [5]. However, sensor accuracy issues in harsh outdoor environments reduced the system's reliability.

Bhargavi and Pachipala (2023) proposed a claims-based identity management paradigm for security in IoT, which unites blockchain, cryptographic encryption, multi-factor authentication, and access control mechanisms [6]. It's strengths of the work entail security enhancement, decentralized identity verification and scalability. The limitations are high computing demands, complex implementation, network dependence, latency involvement, and the challenges of user adaptation.

Kim and Lee (2022) developed an AI-powered IoT system that uses machine learning to classify different noise types (e.g., traffic, industrial, human speech), enabling automatic identification of noise sources, enhancing data accuracy for noise regulation authorities [8]. However, its high processing demands made it incompatible with low-cost microcontrollers like the ESP8266.

Patel and Shah (2021) sketch out an IoT- and cloud-based system for monitoring and controlling noise pollution. It uses wireless sensor networks, machine learning, edge computing, and GIS for online data collection and predictive analysis [10]. Their strongholds include real-time monitoring of the noise pollution levels, high scalability, AI-driven noise control, access to the information through the cloud, and automated noise regulation. Its weaknesses include high expenses incurred in setting it up, network and power dependency, data security risks, high maintenance of sensors, and potential latency issues.

Sahu and Sahoo (2021) have proposed an AI-integrated IoT system for predictable environmental noise pollution, using machine learning, cloud computing, GIS, and wireless sensor networks [11]. Its strengths include real-time monitoring, predictive noise control, scalability, remote accessibility, and automated anomaly detection, while limitations involve high computational demands, network dependency, data security risks, sensor maintenance, and implementation costs.

Zhao and Huang (2021), in Smart Noise Control System Using Wireless Sensor Networks (WSN), used a distributed network of noise sensors communicating with a central cloud platform for real-time analysis and provided comprehensive urban noise maps, useful for government noise control policies [12]. It has faced data transmission issues in dense urban areas due to low signal strength.

Fernando and Gomez (2021), an IoT-Driven Noise Pollution Monitoring Platform for Smart Cities and implemented city-wide IoT noise sensors that upload data to centralized cloud platforms for urban noise tracking [13]. Helped city planners understand long-term noise pollution trends. However, it is limited to high data latency in large-scale deployments due to overloaded cloud servers.

Ahmed and Raza (2021) created an industrial noise monitoring system wherein a Raspberry Pi acts as a nodal point, with sound data being collected from various sound sensors [14]. This design permits the monitoring of noise patterns; however, it is very much dependent on permanent network connectivity, making it unreliable in areas with poor internet access.

Choi and Park (2020): IoT-Based Smart Noise Monitoring for Residential Areas. It is a real-time noise tracking system using Wi-Fi sensors for monitoring residential environments [18]. It is enabled for homeowners to receive alerts when noise exceeded acceptable levels. It requires difficult calibration for several indoor environments.

Singh and Kaur (2019) introduced an IoT-based smart city noise monitoring system, integrating IoT sensors and web-based data transmission [22]. Although the system provided valuable noise level trends, it struggled with high traffic during operations, leading to delays in real-time updates.

Developed an IoT-based noise monitoring system using ESP8266 and Arduino to transmit real-time noise data to a cloud server [23] [26]. While their solution was inexpensive and easy to deploy, upscaling problems were encountered in a large urban context.

In Table I, the literature review presents suggested system improvements; the main aim is to provide a comprehensive analysis of noise levels and their correlation with alert triggers across different environments and time intervals.

The study focuses on understanding noise variations, identifying patterns, and offering data-driven insights to optimize noise management strategies. Compared to these solutions, our proposed system provides real-time alerts, an intuitive web-based dashboard, AI-driven noise classification, and smart home integration, making it more scalable and adaptable for both residential and institutional environments.

TABLE I LITERATURE REVIEW

Author	*Technology Used*	*Strengths*	*Limitations*	*How Proposed System Improves*
Verma & Sharma (2023) [5]	WiFi-enabled sensors, Web interface.	Affordable, easy deployment in public spaces.	Low sensor accuracy due to environmental interference.	Introduces AI-based noise filtering & calibration for better accuracy.
Vasilenko, O., & Orlov, A (2020) [15]	IoT, wireless sensor networks, cloud computing.	Energy efficiency, scalability, and remote accessibility.	High costs, data security risks, complex maintenance, and user adaptation challenges.	Auto-calibration for dynamic adjustments.
Patel & Desai (2020) [16]	Arduino-based indoor noise monitor.	Remote accessibility, improved indoor comfort, and automated alerts.	Limited range of detection and more power consumption, and implementation costs.	Multiple sensors allow coverage of large rooms.
Tavakol, M., & Shahrivar, M. (2020) [17]	IoT, wireless sensor networks, cloud computing, AI-driven analytics, and GIS.	Scalability, AI-driven insights, remote accessibility.	High infrastructure costs, data security risks, power and network dependency, and interoperability challenges.	Hybrid local-cloud processing for faster alerts.
Hassan, W. H., & Akram, A. (2020) [19].	Microcontrollers with noise sensors and wireless communication.	Real-time Monitoring, cloud-based accessibility, scalability.	Network dependency, power consumption, and potentially high costs.	Uses lightweight AI models for ESP8266.
Li, K., & Zhang, X. (2019) [20].	Wireless Sensor Network (WSN), cloud computing, big data analytics, edge computing.	Real-time monitoring, scalability, cloud-based accessibility.	Network reliability issues, data security risks, high implementation costs.	Optimized Wi-Fi-based transmission for indoor spaces.
Khan, A., & Javed, M. (2019) [21].	Wireless sensor networks, cloud and edge computing.	Real-time monitoring, scalability, automated noise mitigation, and remote accessibility,	High implementation costs, power consumption, network dependency, data security risks.	Use local & cloud storage for uninterrupted monitoring.
Meyer, E., & Knoblauch, H. (2017) [24].	Integrating IoT-enabled noise monitoring, AI-driven analysis, cloud computing.	AI-driven mitigation, scalability, and remote accessibility.	High costs, power and connectivity dependency, data security risks, complex maintenance.	Uses optimized data processing to prevent lags and ensure real-time alerts.
Kumar, S., & Jain, P. (2017) [25].	Wireless sensor networks, cloud and edge computing.	Remote accessibility, early warning capabilities, and smart city integration.	No real-time alerts, limited scalability, struggle with multiple sensors. high power consumption.	Introduces real-time notifications, web dashboard, and AI-driven noise classification.
Proposed System	ESP8266, Smart Web Dashboard, AI-based Noise Classification, Cloud & Local Storage.	Real-time alerts, AI-driven pattern recognition, Smart Home Integration, Reliable Data Processing.	Future improvements may include deep learning for advanced noise analysis.	Scale, more accurate, integrates AI & mobile alerts, works even in low-connectivity areas.

III. METHODOLOGY

The IoT-based Noise Pollution Monitoring System consists of three core components:

A. Noise Detection Unit:

A sound sensor captures ambient noise levels and sends data to the Arduino microcontroller.

B. Processing and Data Transmission:

The ESP8266 Wi-Fi module transmits real-time data to a web-based server and triggers alerts.

C. User Interface and Alerts:

A web-based dashboard visualizes noise levels, and email alerts are sent when noise exceeds a set threshold.

To simplify the data flow and decision-making process, the following flowchart illustrates how the system functions:

The functional flow of the entire IoT-based noise pollution monitoring system is represented step-by-step in Fig. 1, ranging from the stage of detection to processing and responding in real-time to the noise level variations. It includes the major functional components, decision-making situations, and error-handling methods defining its reliable operation across several environments.

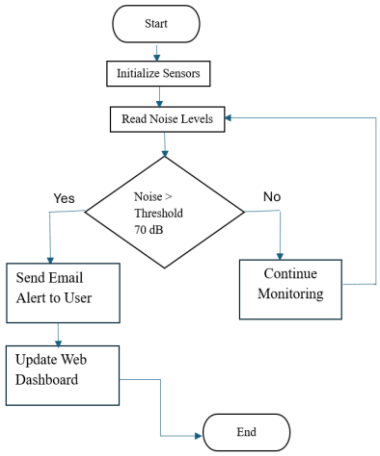


Fig. 1. Flowchart of the proposed IoT-based noise monitoring system.

1) Initialization phase: The initial phase requires the hardware and software algorithms of the system to be initialized:

The sound sensor (KY-038/LM393) is triggered to monitor the environmental noise in real-time. Arduino Uno or the ESP8266-based microcontroller takes care of processing data. The ESP8266 Wi-Fi module gets a connection to establish the network for cloud data transmission. During sensor calibration, reference noise levels are applied to increase the sensor's accuracy. A threshold level of noise is taken at 70dB, upon which the alarms shall be sounded [9]. A moving average filter is put in place to smoothen noise data and reduce false alerts due to brief disturbances.

2) Continuous noise monitoring: The system, once initialized, enters a continuous monitoring loop where it undertakes the following functions:

It will read noise data from the sensor on a regular basis. Moving average smoothing will be applied to reduce random fluctuations in noise levels. The processed noise data will then be classified using an AI-based noise classifier-into normal background noise and serious disturbance.

3) Decision on alert handling: In this phase, the alarm decides whether a noise event needs user intervention or not:

Noise above 70 decibel for 5 seconds: as a sound source, the alarm validates that it is a valid noise or a transient disturbance (e.g., a sudden transient sound, such as door slamming). Once designated valid, an alarm notification is generated by different means; Email alerts to the users on an excessive noise being present. Web-based dashboard updates to give real-time visibility into noise trends. Integration of smart home automation mechanisms that can opt to reduce noise (e.g., close windows, turn down speaker volume). Transient or invalid noise: the alarm has nothing to do but ignore and keeps monitoring.

4) Error handlings and data corrections: For reliability, the system supports error-handling mechanisms capable of dealing with potential failures:

If an error is detected in the sensor (a wrong reading because of a change in the environment), the system self-calibrates based on the changes to adjust the sensor sensitivity. In the event of a network error (Wi-Fi disconnection), the system stores noise data locally, once the network is restored, it attempts to upload the noise data to ensure no data goes lost. The analysis of historical data is used to adjust the noise threshold dynamically, accordingly, to average long-term trends.

The system implements further noise processing techniques to improve accuracy and reduce false alarm rates. Should echo be detected in any enclosed space through continuous reverberation, echo cancellation is performed. Transient noise filtering helps to prevent transient sounds like that of a dropped object or a cough from producing false alarms. Historical noise patterns are subjected to alarm threshold adjustment through online updating based on cumulative environmental experience. Combination of sensor readings, in case of multiple deployments, cross-verification is done to help in maximizing accuracy and preventing erroneous functioning of sensors.

Algorithm 1: Noise Pollution Monitoring with AI-Based Filtering & Error Handling Algorithm

Step 1: Initialization Phase
Step 2: Continuous Noise Monitoring
Step 3: Decision Making & Alert Handling
Step 4: Alert & Data Logging
Step 5: Error Handling & Data Correction

```
Start
Step 1: Initialization Phase
Initialize Sound Sensor (KY-038/LM393),
  Arduino Uno (or ESP8266-based board),
ESP8266 WiFi Module and web server
Calibrate sensor using reference noise levels
    Set noise threshold = 70 dB
Initialize moving average filter
Initialize MQTT for real-time data transmission
```

```
Initialize MQTT for real-time data transmission
Step 2: Continuous Noise Monitoring
        WHILE (true):     // continuously monitor noise levels
  noise_level = Read_Sensor_Data // Read real-time noise level (dB) from the sound sensor.

noise_filtered=Apply_Moving_Average_Filter(noise_level)
  noise_classified = AI_Noise_Classifier(noise_filtered)
Step 3: Decision Making & Alert Handling
  IF (noise_classified > 70 dB) for at least 5 seconds:
      IF Is_Valid_Noise(noise_classified) // Verify if the noise is from a valid source
Step 4: Alert & Data Logging
        Send_Email_Alert(noise_classified)
            Update_Web_Dashboard(noise_classified)
   ELSE:
        Ignore_Transient_Noise ()
Step 5: Error Handling & Data Correction
  IF Sensor_Error ():          // Handle Sensor Errors
      Perform_Self_Calibration ()
  IF Network_Error ():        // Handle Network Delays
        Store_Data_Locally ()
            Retry_Upload ()

Optimize_Threshold_Based_On_Historical_Data ()
End
```

TABLE II HARDWARE COMPONENTS AND FUNCTIONS

Hardware Component	*Functions*
Arduino Uno	Processes sensor data and triggers alerts
Sound Sensor (KY-038/LM393)	Measures ambient noise in decibels (dB)
ESP8266 WiFi Module	Transmitting data to a web dashboard
Web Server (Cloud)	Stores and displays real-time noise data
Email Notification System	Sends alerts when noise exceeds threshold

In Table II, shown on the hardware components and functions, the Arduino Uno Reads noise level from KY-038/LM393 sound sensor. Implements a moving average filter for stability. Sending data to an MQTT broker for real-time monitoring triggers an alert, if noise levels exceed the threshold (70dB), the ESP8266 module transmits data. Alerts are sent via email, and data is displayed on a web interface.

Connectivity: ESP8266, Wi-Fi, and MQTT

The ESP8266 Wi-Fi module is essential for real-time wireless communication in IoT-based noise monitoring systems. It enables the device to connect to the internet, transmit noise data to a server, and receive control commands remotely.

ESP8266 connects to Wi-Fi with a step-by-step procedure.

a) Enables internet-based communication.

b) MQTT (Message Queuing Telemetry Transport): Efficient protocol for real-time data exchange.

c) Noise data is published to an MQTT broker: Cloud platforms (AWS IoT)

d) Web Dashboard subscribes to MQTT: Displays real-time noise levels.

e) Alerts: If noise is too high, an alert is sent via MQTT to a buzzer, email.

IV. RESULT AND ANALYSIS

The proposed work is tested over a seven-day period in different environments, including offices, residential areas, and classrooms.

Table III presents the noise levels recorded at different time intervals over a seven-day period, along with whether an alert was triggered. The data shows that noise levels exceeding 70dB consistently triggered alerts, with peak values occurring between 10:50 AM and 5:30 PM. Lower noise levels were recorded during early mornings and late evenings, highlighting a direct correlation between human activity and noise pollution.

The data confirms that noise levels above 70dB consistently trigger alerts, with peak noise occurring during busy hours (10:50 AM to 5:30 PM). The highest recorded noise level was 98dB, which is significantly above the recommended limit for indoor environments. Early morning and late evening noise levels remained below 60dB, indicating quieter periods. These patterns suggest the need for adaptive noise control measures in high-activity periods.

TABLE III A SEVEN-DAY PERIOD IN NOISE LEVELS AND ALERTS OVER TIME

Time Interval	*Noise Level*	*Alert Triggered*
8:30AM – 9:00 AM	76dB	Yes
9:30 AM- 9:50AM	60dB	No
10:50AM-11:00AM	85dB	Yes
12:40PM- 1:30PM	80dB	Yes
3:10PM- 3:20PM	92dB	Yes
5:20PM-5:30PM	98dB	Yes
6:00PM	52dB	No

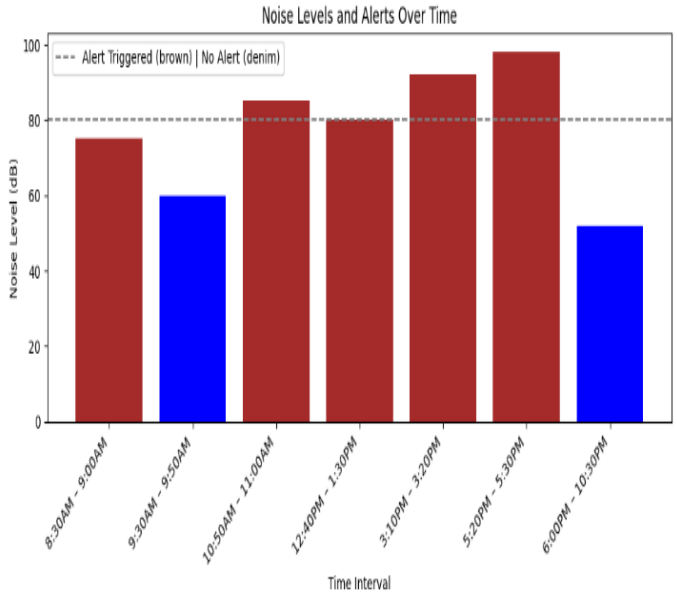


Fig. 2. A seven-day period: Noise levels and alerts over time.

Fig. 2 represents noise levels over time. The Dashed Line shows an 80dB threshold for better visualization. The X-axis denote the time interval; the Y-axis represents the noise level (dB). Each color corresponds to the noise level for a given period.

Color Coding

- Brown → Alert Triggered (≥ 80dB)
- Blue→ No Alert (< 80dB)

Peak noise levels (above 80 dB) were observed between 10:50 AM and 5:30 PM, likely due to increased activity in workplaces and classrooms.

Lower noise levels: (below 60 dB) occurred in the early morning and late evening, when fewer people were present.

Unexpected fluctuations: (75 dB at 8:30 AM) suggest temporary noise sources, possibly from nearby construction or sudden disturbances.

The system was deployed in three different environments – a home, an office, and a classroom for one week. The noise

levels were continuously monitored, and alerts were triggered whenever the threshold exceeded 70dB.

Table IV summarizes noise levels recorded over a week in three different environments: home, office, and classroom. Offices had the highest noise levels, often exceeding 85dB, likely due to conversations, machinery, and workplace activities. Classrooms showed moderate noise peaks, particularly during break times and group discussions, while home environments generally had lower noise levels but occasional spikes due to appliances and social interactions.

TABLE IV WEEKLY NOISE LEVEL

Day	*Noise Level*	*No. of Alerts Triggered*	*Environment*
Sunday	76dB	2	Home
Monday	89dB	3	office
Tuesday	77dB	4	Classroom
Wednesday	90dB	6	office
Thursday	87dB	4	Classroom
Friday	91dB	3	Home
Saturday	86dB	2	Home

The results highlight that office environments are the noisiest, frequently exceeding 85dB, which could impact productivity and health. Classrooms experience moderate noise fluctuations, likely due to student activities, while homes have generally lower noise levels but occasionally high readings (e.g., 91dB on Friday). This suggests that custom noise mitigation strategies are needed for different environments, such as soundproofing offices and using AI-based filtering for home settings.

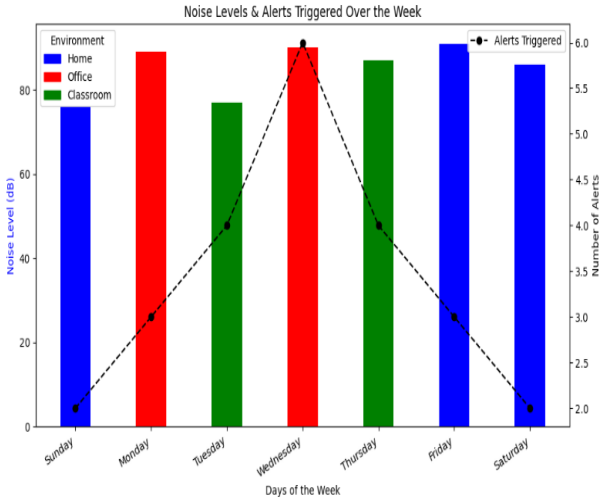


Fig. 3. Noise level and alerts triggered over the week.

In Fig. 3, the X-axis signifies Days of the Week, Y-axis denotes Noise Level (dB). Each color corresponds to the noise level for a given day. Use different colors for different environments (Home, Office, and Classroom) for better visualization. The dual Y-axis shows one for Noise Level, and one for Alerts.

Office Environments: Had the highest noise spikes (85 to 90 dB) due to frequent conversations, phone calls, and background noise machinery.

Classrooms: Exhibited moderate peaks (77 to 87 dB), mainly during break times and group activities. Homes: Showed the least noise disturbances, but occasional high readings (up to 91 dB) were detected, possibly due to TVs, kitchen appliances, and social interactions.

1) Data analysis: At Home, Noise levels are moderate (Sunday: 76dB, Friday: 91dB, Saturday: 86dB). At Office, the highest noise levels occur here (Monday: 89dB, Wednesday: 90dB), leading to more alerts. The Classroom shows consistent noise levels (Tuesday: 77dB, Thursday: 87dB), with a moderate number of alerts. Office environments tend to be the loudest, with noise exceeding 89dB, which can impact productivity and health. Home environments have moderate noise levels, but Friday shows an unexpected spike (91dB). Possible causes should be investigated. Classroom noise is relatively stable, but alert triggers suggest some noise concerns on specific days (Tuesday and Thursday). Wednesday is the noisiest day overall (90dB, 6 alerts), indicating a need for noise control measures.

2) Mean (Average): Represents the overall noise level trend.

$$Mean = N/\sum X \tag{1}$$

3) Median: Shows the middle noise value, reducing the impact of extreme values.

$$median = \frac{N+1}{2} \tag{2}$$

$$minimum = \min(x, x2, x3, \dots, xn) \tag{3}$$

$$maximum = \max(x1, x2, x3, \dots, xn) \tag{4}$$

4) Standard Deviation: Measures how much the noise levels fluctuate.

$$\sigma = \sqrt{\frac{\sum(xi - \bar{x})2}{N}} \tag{5}$$

where,

$\sum X$ = Sum of all values

N = Number of values

σ = Standard deviation

Xi = Each data point

$\bar{X}$ = Mean (average) of the dataset

x =represents the values in the dataset.

To better understand noise variations, Table V presents key statistical metrics, including the mean, median, and standard deviation of noise levels and alerts triggered. The mean noise level (85.14dB**)** suggests consistently high noise across environments, while a strong positive correlation (+0.85) indicates that as noise levels rise, the number of alerts increases. These insights help with fine-tuning alert thresholds and noise mitigation strategies.

TABLE V STATISTICAL ANALYSIS OF NOISE LEVELS AND ALERTS

Metric	*Noise Level(dB)*	*Alerts Triggered*
mean	85.14	3.43
Median	87	3
Minimum	76	2
maximum	91	6
Stand deviation	5.28	1.51

The statistical analysis supports the observed trends: the high mean noise level (85.14dB) and strong correlation (+0.85) confirm that increasing noise levels directly lead to more alerts. The standard deviation (5.28dB) indicates moderate fluctuations in noise levels across different environments. These insights validate the system's ability to accurately detect and respond to excessive noise, making it a reliable solution for noise monitoring.

Correlation Analysis: Correlation (+0.85) indicates a strong relationship between noise levels and alert triggers. As noise levels increase, the number of alerts triggered also rises. This suggests a direct relationship, where higher noise levels frequently cross the threshold for triggering alerts.

$$r = \frac{N\sum(X_iY_i) - \sum X_i \sum Y_i}{\sqrt{[N\sum X_i^2 - (\sum X_i)^2] \times [N\sum Y_i^2 - (\sum Y_i)^2]}}$$

r = Pearson correlation coefficient

Xi = Values of the first variable (e.g., Noise Levels)

Yi = Values of the second variable (e.g., Alerts Triggered)

N = Number of data points

∑Xi = Sum of all X values

∑Yi = Sum of all Y values

∑(XiYi) = Sum of the product of each pair of X and Y

∑Xi2 = Sum of squares of X values

∑Yi2 = Sum of squares of Y values

Were

r=1 Perfect positive correlation (both variables increase together)

r=−1 Perfect negative correlation (one increases, the other decreases)

r=0 No correlation (variables are unrelated)

0.7≤r<1 Strong positive correlation

0.3≤r<0.7 Moderate positive correlation

0≤r<0.3 Weak positive correlation

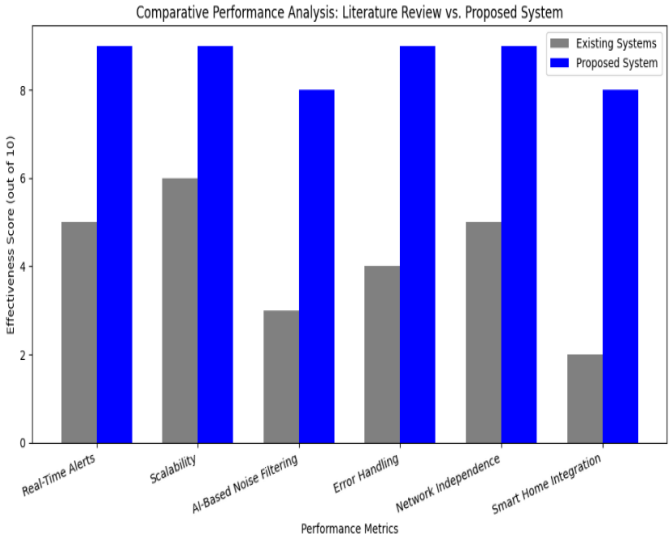


Fig. 4. Comparative performance analysis.

In Fig. 4, the performance comparison highlights the superiority of the proposed system over existing noise monitoring solutions. It effectively addresses real-time alert limitations, AI-based noise classification, and offline functionality, ensuring greater adaptability and reliability. Unlike traditional systems that struggle with scalability and processing power, the proposed approach offers enhanced accuracy and real-time responsiveness, making it an ideal choice for diverse environments.

Table VI highlights the limitations found in previous noise monitoring studies and how the proposed system overcomes these challenges. Key improvements include real-time alerts, AI-based noise classification, and offline functionality, making the system more adaptable and reliable across different environments. By addressing these gaps, the system provides a more effective solution for noise pollution management.

TABLE VI PROPOSED SYSTEM ADDRESSES EXISTING GAPS

Existing Limitations (from Literature Review)	*Proposed Work Solution*
Lack of real-time alerts	Uses MQTT messaging, emails, and web dashboards for instant notifications
Poor scalability for different environments	Works in small-scale (homes/offices) and large-scale (industries/smart cities) deployments
No AI-based noise classification	Uses AI models to filter out background noise & reduce false alerts
High dependency on stable internet	Works offline by storing data locally & syncing it later
No smart home integration	Connects with smart IoT devices to control noise-sensitive appliances

Table VI represents how the proposed system addresses existing gaps. The proposed work ensures a real-time, intelligent, and scalable noise monitoring solution that overcomes limitations of existing research by integrating AI, IoT, smart automation, and error-handling techniques for improved noise pollution management.

The proposed system significantly improves upon existing solutions by offering real-time alerts, AI-based noise filtering, and offline functionality, which were missing in previous studies. Unlike traditional systems that struggle with scalability and accuracy, this system ensures efficient noise classification and adaptive monitoring across different environments. These enhancements make it a versatile and practical approach to noise pollution control.

V. Conclusion

The proposed IoT-based noise pollution monitoring system effectively addresses the key limitations found in existing research by integrating real-time noise monitoring, AI-based noise classification, smart home automation, and offline functionality. Through comparative analysis with previous studies, it is evident that this system provides superior performance in terms of accuracy, scalability, efficiency, and adaptability.

This study confirms that AI-integrated IoT systems enhance noise detection, reduce false alarms, and improve alert response time, while offering a scalable solution for various environments. Additionally, the system's offline capabilities ensure continuous monitoring, making it highly reliable even in low-connectivity areas.

References

[1] L.-J. Chen, S. Saraswat, F.-S. Ching, C.-Y. Su, H.-L. Huang, and W.-C. Pan. “Development and implementation of EcoDecibel: A low-cost and IoT-based device for noise measurement”, Ecological Informatics (2025), vol. 85, p. 102968.

[2] Srikar, S.S., Yellamma, P. “An IoT-based Intelligent Irrigation and Weather Forecasting System”, Recent Patents on Engineering (2024), 18 (9), pp. 100-108.

[3] How much does environmental noise affect our health? WHO updates methods to assess health risks, https://www.who.int/europe/news/item/04-08-2024-how-much-does-environmental-noise-affect-our-health--who-updates-methods-to-assess-health-risks

[4] Mopuru, B., Pachipala, Y. “Advancing IoT Security: Integrative Machine Learning Models for Enhanced Intrusion Detection in Wireless Sensor Networks”, Engineering, Technology and Applied Science Research (2024), 14 (4), pp. 14840-14847.

[5] Verma, M., & Sharma, P. “Low-Cost IoT-based Noise Monitoring System for Public Spaces”. International Journal of Smart Cities and Systems Engineering (2023), 12(4), 93-102.

[6] Bhargavi, M., Pachipala, Y. “Enhancing IoT Security and Privacy with Claims-based Identity Management”, International Journal of Advanced Computer Science and Applications (2023), 14 (11), pp. 822-830.

[7] Cheng, H., & Li, Q. “AI-based Noise Detection and Classification in Smart Homes”, Journal of Smart Home Technology (2022), 30(3), 276-289.

[8] Kim, D., & Lee, S. (2022). “AI-Powered IoT System for Noise Classification and Source Identification”, IEEE Access, 10, 65012-65023.

[9] WHO Compendium on Health and Environment – Environmental Noise (2022),https://cdn.who.int/media/docs/default-source/who-compendium-on-health-and environment/who_compendium_noise_01042022.pdf?sfvrsn=bc371498_3

[10] Patel, S., & Shah, V. “Noise Pollution Monitoring and Control Using IoT and Cloud Computing”, IEEE Internet of Things Journal (2021), 8(4), 2557-2567.

[11] Sahu, P., & Sahoo, G. “Integration of IoT with AI for Predictive Environmental Monitoring: A Case Study on Noise Pollution”, Journal of Artificial Intelligence and Environmental Technology (2021), 18(4), 102-118.

[12] Zhao, X., & Huang, T. “Smart Noise Control System Using Wireless Sensor Networks”, Journal of Urban Technology (2021), 28(2), 25-37.

[13] Fernando, M., & Gomez, L. “IoT-Driven Noise Pollution Monitoring Platform for Smart Cities”, Urban Climate (2021), 39, 100933.

[14] Ahmed, M., & Raza, S. “Industrial Noise Monitoring Using Raspberry Pi and IoT”, Journal of Industrial Engineering & Management (2021), 14(2), 109-118.

[15] Vasilenko, O., & Orlov, A. “Artificial Intelligence for Smart Home Automation and Environment Monitoring”, Journal of Ambient Intelligence and Smart Environments (2020), 12(6), 85-103.

[16] Patel, S., & Desai, V. “IoT-Based Noise Monitoring for Indoor Spaces”, Journal of Smart Environments and Sustainability (2020), 4(1), 15-24.

[17] Tavakol, M., & Shahrivar, M. “Real-time Environmental Monitoring Systems: Applications in Smart Cities”. Journal of Urban Technology (2020), 27(2), 1-19.

[18] Choi, J., & Park, J. “IoT-Based Smart Noise Monitoring for Residential Areas”, International Journal of Smart Home (2020), 14(2), 56-67.

[19] Hassan, W. H., & Akram, A. “Real-time Smart Noise Monitoring System Using IoT with Cloud-based Visualization”. International Journal of Environmental Research and Public Health (2020), 17(16), 5671.

[20] Li, K., & Zhang, X. “IoT-based Real-time Environmental Monitoring: Noise Pollution Case Study”, IEEE Access (2019), 7, 125722-125732.

[21] Khan, A., & Javed, M. “A Smart IoT System for Noise Pollution Monitoring and Control in Smart Cities”. Sensors (2019), 19(22), 4922.

[22] Singh, R., & Kaur, P. IoT-based Smart City Noise Monitoring System. International Journal of Computer Science and Engineering Technology (2019), 10(3), 149-157.

[23] Gautam, S., Kumar, A., & Patil, S. “IoT-based Noise Pollution Monitoring System Using ESP8266 and Arduino”, International Journal of Advanced Research in Electrical, Electronics, and Instrumentation Engineering (2018), 7(7), 2736-2743.

[24] Meyer, E., & Knoblauch, H. “Noise Pollution Control: Traditional Methods vs. Smart Technologies”, Environmental Science & Technology (2017), 51(4), 1201-1211.

[25] Kumar, S., & Jain, P. “IoT-based environmental monitoring system for real-time monitoring and control of air quality”, Journal of Environmental Science and Technology (2017), 45(3), 43-56.

[26] Parupally Venu, Pachipala Yellamma*, Yama Rupesh, Yerrapothu Teja Naga Eswar, Maruboina Mahiddar Reddy.” Dynamic Priority-Based Round Robin: An Advanced Load Balancing Technique for Cloud Computing”, International Journal of Advanced Computer Science and Applications, Vol. 15, No. 9, 2024, pp 252-258.

Deep Learning Meets Bibliometrics: A Survey of Transfer Learning Techniques for Breast Cancer Detection

Amna Wajid[1], Natasha Nigar[2]*, Hafiz Muhammad Faisal[3], Olukayode Oki[4], Jose Lukose[5]
Department of Computer Science, University of Engineering and Technology, Lahore, Pakistan[1,2]
Technical University of Munich, Munich, Germany[3]
Department of Computer Science, Walter Sisulu University, South Africa[4,5]

***Abstract*—This study aims to provide a comprehensive bibliometric analysis of research on transfer learning in breast cancer detection from 2016 to 2024. It highlights publication trends, influential contributors, collaborations, and keyword patterns. Bibliometric methods are employed to analyze data extracted from the Scopus database. It includes co-occurrence and citation analyses to identify prevalent keywords, highly cited documents, journals, authors, organizations, and countries contributing to this field. The analysis reveals a significant upward trend in publications over the last decade. Key insights include the identification of dominant keywords, influential contributors, and notable collaborations. The results highlight the growing impact of transfer learning techniques in breast cancer detection research, particularly within the domains of medical imaging analysis and predictive analysis. This study offers a systematic overview of the current state of transfer learning in breast cancer detection research, providing valuable insights and guiding future research efforts in this rapidly evolving domain.**

***Keywords*—Transfer learning; breast cancer; medical imaging analysis; predictive analysis**

I. Introduction

The breast cancer remains a significant global health concern. This is the most commonly diagnosed cancer among women and is a leading cause of cancer-related deaths worldwide [1]. Early detection and an accurate prognosis of breast cancer are crucial to improving patient outcomes [2]. There are significant advancements in medical imaging technologies such as mammography, magnetic resonance imaging (MRI), breast ultrasound and thermography. However, multiple challenges exists in manual interpretation of the generated images that leads to variations in diagnosis accuracy among clinicians [3]. Many researchers proposed methodologies based on artificial intelligence (AI) and deep learning (DL) to enable early stage breast cancer detection to mitigate associated mortality rates. However, these approaches encounter various challenges such as limited access to large datasets, data imbalance and overfitting issues. To tackle these obstacles, transfer learning, in conjunction with machine learning have been employed to increase the efficiency and accuracy of breast cancer classification.

The Transfer Learning (TL) has gained attention for its potential to address challenges in medical image analysis, such as limited dataset sizes and class imbalance issues [4]. This involves leveraging pre-trained models on diverse datasets to improve the performance of breast cancer detection algorithms, particularly in scenarios where large, annotated datasets are unavailable [5].

Despite the fact that TL-based techniques enhance breast cancer detection and survival prediction, there is a gap in the literature regarding bibliometric analyses specifically focusing on TL based approaches. Therefore, understanding the current research landscape and identifying future research trajectories in this domain is essential for advancing breast cancer diagnosis and prognosis. In this context, the proposed bibliometric study aims to address this gap by conducting a comprehensive analysis of research trends and developments in transfer learning-based breast cancer detection techniques. The study focuses on publications, covering the period from 2016 to 2024 by employing the data from Scopus database. This study intends to provide valuable insights into the current state and future directions of research by analyzing the publication trends, influential authors, institutions, journals and keyword patterns. Based on this analysis, researchers and practitioners may gain a deeper understanding of the landscape of transfer learning-based approaches for breast cancer detection, facilitating informed decision-making and guiding future research efforts. In order to fulfill the objectives, this study aims to answer the following research questions:

- RQ1. What is the volume of research on breast cancer detection using transfer learning?
- RQ2. What are the collaboration and authorship patterns of breast cancer research using transfer learning?
- RQ3. Which are the active institutions, authors and journals that are producing breast cancer research using transfer learning?
- RQ4. What are the most frequently used keywords and themes of breast cancer research using transfer learning?
- RQ5. What kind of bibliographic coupling of countries, journals and authors exist in breast cancer research using transfer learning?

The structure of this research is organized as follows: The next section presents a comprehensive literature review, providing an overview of previous bibliometric studies relevant to the topic. This is followed by the materials and methods section,

*Corresponding authors.

which details the method employed to conduct this bibliometric study. The results section then reports the statistical and network analysis of the studies selected for this bibliometric analysis. Subsequently, the discussion section interprets the results. The research then addresses the study's limitations and future directions, identifying potential constraints of the current bibliometric study and proposing areas for further investigation. Finally, the conclusion section summarizes the main findings, highlighting the study's contributions and significance of the research.

II. Literature Review

Ahmad et al. [6] have performed bibliometric analysis of breast cancer research in Pakistan up to Feb, 2021 by utilizing the Web of Science database. The study included 1,605 publications authored by 7,774 individuals, with an average of 18.25 citations per publication. Despite international collaborations with 88 countries, the research output doesn't align with the escalating breast cancer cases in Pakistan, indicating the urgency for policy reassessment and increased attention at the national level to combat the disease effectively.

Syed et al. [7] have utilized bibliometric analysis to assess the evolution of Artificial Intelligence (AI) in breast cancer diagnosis and prognosis research from 2000 to 2021. The analysis identified a significant increase in relevant literature, with the USA, China and India leading in productivity. Key authors, affiliations and journals are highlighted, with trends such as transfer learning and deep learning identified. The findings offer insights for policymakers and researchers to guide future collaborations and advancements in AI for breast cancer care [7].

Khairi et al. [8] have employed bibliometric analysis to investigate the development of deep learning for breast cancer image classification from 2014 to 2021, aiming to address the limitations of manual histopathology examination. Utilizing Scopus data, it examined publication trends, international collaborations and emerging research directions through visualization techniques. The research domain, initiated in 2016, exhibited sustained growth, with notable collaboration between the United States and China. This review aimed to guide researchers by identifying emerging areas in medical imaging for breast cancer diagnosis [8].

Zhou et al. [9] have conducted a comprehensive bibliometric analysis of artificial intelligence (AI) applications in histopathological image (HI) analysis from 2001 to 2021, utilizing data from the Web of Science Core Collection. It revealed a rapid increase in publications, with the USA leading in productivity and influence. Keyword and co-citation analyses highlighted breast cancer, prostate cancer and other tumor types as focal points, with classification and nucleus segmentation emerging as primary research directions. The study offered valuable insights for researchers, identifying key cancer types and trends in AI application in HI [9].

Joshi et al. [10] have performed a quantitative bibliometric analysis of research trends in breast cancer detection using machine learning techniques, focusing on histopathological modalities. Data sourced from Scopus database encompass influential authors, research institutes, journals, highly cited articles and keyword trends. The analysis suggested promising avenues for future research in breast cancer care through machine learning or deep learning methodologies [10].

TABLE I. Summary of Bibliometric Studies on Breast Cancer Detection

Study	Year.	Database	Scope of Study	Duration
Ahmad et al. [6]	2021	WOS	Pakistan	Start-2021
Syed et al. [7]	2022	Scopus, WOS	General	2000 to 2021
Khairi et al. [8]	2021	Scopus	Histopathological Images (HI)	2013-2021
Zhou et al. [9]	2022	WOS	HI Analysis	2001-2021
Joshi et al. [10]	2021	Scopus	HI Analysis	2009-2021

The studies outlined in Table I primarily focuses on bibliometric analysis related to breast cancer detection, with a predominant emphasis on histopathological images. Only one study addresses AI-based breast cancer detection techniques indicating a gap in the literature regarding bibliometric analysis specifically targeting transfer learning-based methods. Transfer learning has emerged as a valuable tool in machine learning, offering the potential to boost the effectiveness of breast cancer detection algorithms by leveraging knowledge from one task to enhance performance in another.

III. Materials and Methods

A. Methodology

Many researchers start their research with bibliometric analysis, which entails quantitative and statistical evaluation of data to assess the research performance of publications and discern trends in research publications across institutions, countries, individuals and disciplines [11]. As compared to other methods of reviewing literature, bibliometric offers a systematic, easy-to-understand and reproducible approach. This study employed bibliometric methods, which involve analyzing various scientific research publications such as research articles, books, conference papers and journals to comprehensively examine the landscape of research on the topic of Transfer Learning in Breast Cancer Detection. Through this bibliometric analysis, the aim is to evaluate the evolution of research in AI and transfer learning techniques for breast cancer detection, identify influential authors, institutions and journals in the field, track publication trends over time and uncover patterns of collaboration and citation among researchers and institutions.

B. Source of Data

Commonly used databases such as Scopus, Web of Science, Scimago and Google Scholar are utilized to gather information about publications and their citations. Among these, Scopus, an academic abstracts and indexing database by Elsevier, stands out for its comprehensive coverage, particularly in health-related topics, surpassing other databases like PubMed and Web of Science [12]. Therefore, for this bibliometric analysis the SCOPUS database is employed to compile data on publications related to breast cancer detection using Transfer Learning, along with additional keywords from 2016 to 2024.

C. Search Query and Data Retrieval

A thorough search strategy, as illustrated in Fig. 1, is implemented to retrieve the results.

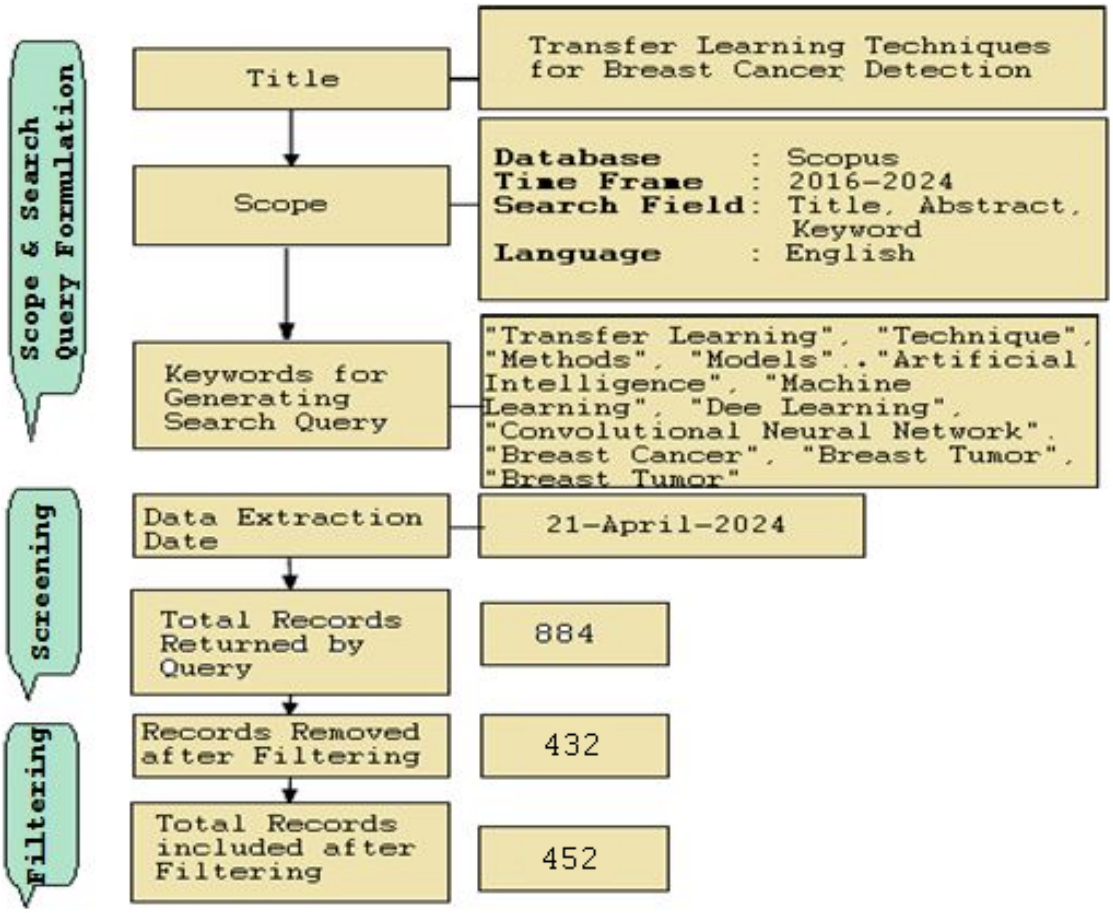


Fig. 1. Search strategy and data filtering.

The literature search is conducted on April 21, 2024. Relevant keywords are identified after consulting pertinent literature and the search query is formulated. These keywords are then combined using Boolean search operators to maximize the retrieval of relevant results. The following search query (Table II) is composed and executed within the SCOPUS database.

TABLE II. QUERY FOR TRANSFER LEARNING TECHNIQUES IN BREAST CANCER DETECTION

TITLE-ABS-KEY (“Transfer Learning”)
AND
TITLE-ABS-KEY (technique* OR method* OR model*)
AND
TITLE-ABS-KEY (“Artificial Intelligence” OR “Machine Learning” OR “Deep Learning” OR “Convolutional Neural Network”)
AND
TITLE-ABS-KEY (“Breast Cancer” OR “Breast Tumor” OR “Breast Carcinoma”)

D. Inclusion and Exclusion Criteria

1) Initial search and query: A search query is conducted on the Scopus database, yielding a total of 884 documents.

2) Screening based on title and abstract: Documents are screened based on their title and abstract, resulting in the retention of 458 articles after filtering.

3) Filtering based on document type: Subsequent filtration is performed to exclude documents that did not meet the specified document types. As a result, 455 records are included for analysis, encompassing journal articles (n = 265), reviews (n = 8), conference papers (n = 175), and book chapters (n = 7), ultimately excluding an additional three papers.

4) Language filtration: A further filtration process is applied to include only articles written in the English language. This resulted in the exclusion of 3 more articles written in Chinese (n=2) and Turkish (n=1) language. After applying these inclusion and exclusion criteria, a total of 452 articles remained for final analysis.

E. Tools Used

The data is processed, analyzed, and visualized using the R-Studio and VOSViewer software.

IV. RESULTS AND DISCUSSION

A. Statistical Analysis

1) Documents by publication years: The analysis of the year-wise trend of publications since last nine years reveals noteworthy insights into the evolution of research in the field. As evident in Table III and shown in Fig. 2, the data presents

a clear upward trend in the yearly publication of transfer learning-based breast cancer detection techniques from 2016 to 2024.

TABLE III. NUMBER OF PUBLISHED ARTICLES BY YEAR

Year	Published Articles	Year	Published Articles	Year	Published Articles
2016	3	2017	9	2018	23
2019	27	2020	39	2021	70
2022	100	2023	133	2024	48

In 2016, only 3 articles published on this topic, indicating a relatively low level of research activity. However, by 2018, the number of publications had more than tripled to 23, signaling a growing interest in transfer learning techniques for breast cancer detection. The trend continued to accelerate in subsequent years, with substantial increases in publication numbers. By 2021, there are 70 articles published, demonstrating a significant surge in research output. This trend reached its peak in 2023, with a staggering 133 articles published, reflecting a flourishing period of research activity in this field. In 2024, up till now 48 articles have been published, indicating a sustained interest in transfer learning-based breast cancer detection techniques.

Overall, the data highlights the increasing prominence of transfer learning-based approaches in breast cancer detection research with a consistent upward trajectory in publication numbers over the specified time period.

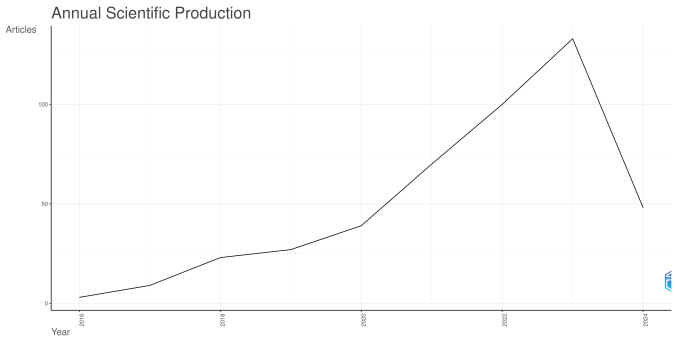


Fig. 2. Yearly publication trends of documents.

2) Documents by average citation per year: The mean total citations per year show a clear declining trend from 2016 to 2024, as shown in Table IV and in Fig. 3. In 2016, the mean total citations per year are 25.52, indicating a relatively high citation rate. However, this number steadily decreased over the subsequent years. By 2024, the mean total citations per year plummeted to just 0.42, suggesting a significant decline in the citation impact of the research publications. This downward trend may reflect a shift in research focus, changes in citation practices, or evolving trends in the academic landscape.

3) Documents by author: The data on the top 10 most relevant authors in breast cancer detection research using transfer learning reveals a diverse pool of contributors. Fig. 4 shows the 10 most relevant authors in this domain. Maryellen L. Giger emerges as the most prolific author with 10 articles to her name, indicating a significant contribution to the field. Mustapha Bouakkaz, Saida Sarra Boudouh, and Se-Woon Choe

TABLE IV. AVERAGE CITATIONS PER YEAR

Year	MeanTCperArt	N	MeanTCperYear	CitableYears
2016	229.67	3	25.52	9
2017	128.33	9	16.04	8
2018	61.87	23	8.84	7
2019	56.19	27	9.37	6
2020	38.97	39	7.79	5
2021	24.30	70	6.08	4
2022	11.88	100	3.96	3
2023	3.73	133	1.86	2
2024	0.42	48	0.42	1

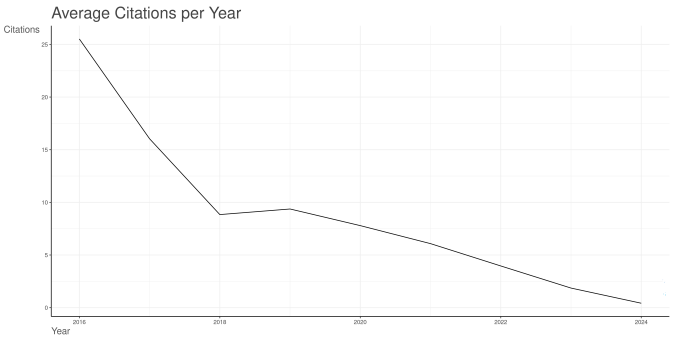


Fig. 3. Average citations per year.

follow closely behind with 7 articles each, underscoring their substantial involvement in research publications. Gelan Ayana and Sannasi Chakravarthy, S.R. are also notable contributors with 6 articles each. The remaining authors, including N. Bharanidharan, Hui Li, My Hachem El Yousfi Alaoui, and Mohamed Bal-Ghaoui, have contributed 5 or 4 articles each, demonstrating their significant engagement in research endeavors related to breast cancer detection using transfer learning techniques.

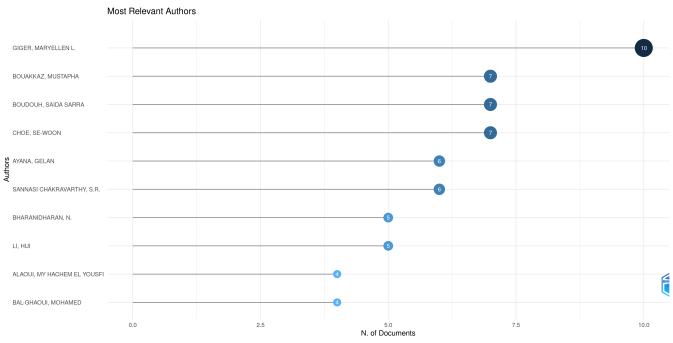


Fig. 4. Documents by relevant authors.

4) Documents by affiliations: The data on the top 10 affiliations involved in research on breast cancer detection using transfer learning reveals a diverse range of institutions contributing to this field. “Princess Nourah bint Abdulrahman University” emerges as the top affiliation with 24 articles, indicating its significant involvement and leadership in research endeavors related to breast cancer detection. Following closely behind is “I.R.C.C.S. Istituto Tumori ‘Giovanni Paolo

II" with 22 articles, highlighting its substantial contribution to the scientific literature in this area. "Harvard Medical School", "Kumoh National Institute of Technology", and "University of Michigan", each have 16 articles, showcasing their active engagement in breast cancer detection research. Similarly, "University of Rome Tor Vergata", "Icahn School of Medicine at Mount Sinai", "King Abdul Aziz University", "University of California", and "COMSATS University Islamabad (CUI)", each contribute significantly with 15, 13, 13, 13, and 12 articles respectively, underscoring their importance as key players in advancing knowledge and innovation in breast cancer detection using transfer learning techniques [see Fig. 5].

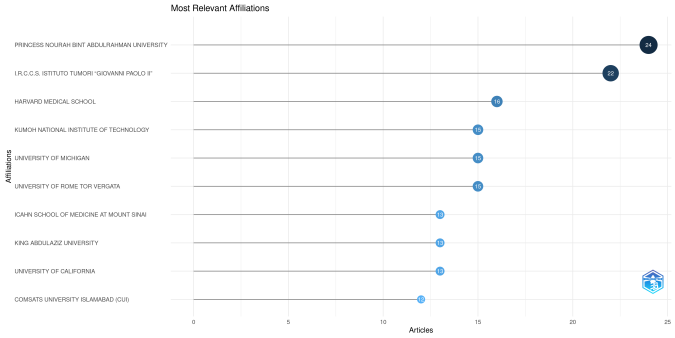


Fig. 5. Documents by affiliations.

5) Documents by sources: The data on the most relevant sources highlights a diverse range of publication venues. "Progress in Biomedical Optics and Imaging - Proceedings of SPIE" emerges as the top source with 18 articles, indicating its significant contribution to the dissemination of research in this field. Following closely behind are "Multimedia Tools and Applications" and "Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)" with 14 and 13 articles, respectively, underscoring their importance as platforms for scholarly communication. "Diagnostics", "Biomedical Signal Processing and Control", "Cancers", and "Computers in Biology and Medicine" , each contribute substantially with 11, 10, 9, and 9 articles, respectively, indicating their relevance in the field. Other notable sources include "IEEE Access", "Communications in Computer and Information Science", and "Lecture Notes in Networks and Systems", each with 9, 8, and 7 articles, respectively, reflecting their role in disseminating research findings on breast cancer detection using transfer learning techniques [see Fig. 6].

6) Documents by countries: The data reveals a varied landscape of research contributions in breast cancer detection using transfer learning techniques across different countries. China and India emerge as the most prolific contributors, with 35 and 90 articles, respectively, showcasing their significant involvement in this field. Other countries, such as Egypt, also demonstrate substantial participation with 11 articles. Interestingly, while some countries like China and India predominantly contribute through single-country publications (SCP), others like Canada and Italy engage in both SCPs and multiple-country publications (MCPs), indicating diverse collaboration patterns. This highlights the global nature of research efforts in breast cancer detection, emphasizing the importance of international cooperation to advance scientific knowledge and

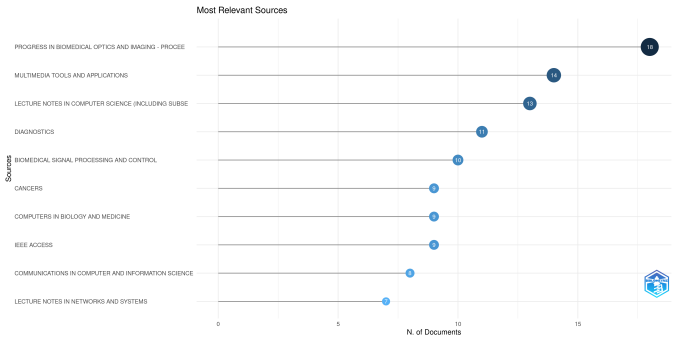


Fig. 6. Documents by sources.

address complex healthcare challenges. The data underscores the need for continued collaboration and knowledge-sharing among researchers worldwide to drive innovation and progress in breast cancer detection methodologies [see Fig. 7].

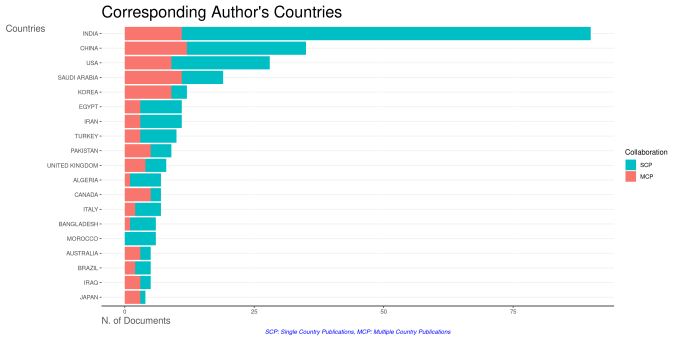


Fig. 7. Documents by countries.

B. Network Analysis

This phase of the study focuses on examining various network metrics, including co-authorship, co-occurrences, citations, bibliographic coupling, and co-citation.

1) Co-authorship analysis: Co-authorship analysis involves examining collaborations among authors, organizations, and countries.

a) Co-authorship analysis with authors: In this analysis, a threshold of at least 2 documents per author is set. Out of 1680 authors, 215 authors met this threshold. The largest connected set comprises 18 out of the 215 authors, as illustrated in Fig. 8.

b) Co-authorship analysis with organizations: The dataset comprises publications affiliated with 1153 organizations. Among these, 48 organizations meet the threshold criteria of having at least 2 documents. However, not all of these 48 entities are interconnected. The largest connected set encompasses 10 items, as illustrated in Fig. 9.

c) Co-authorship analysis with countries: The publications included in this study originate from 73 different countries. Among these, 51 countries meet the threshold requirement of having at least 2 documents. Within these 51 countries, the largest interconnected set comprises 49 items, as depicted in Fig. 10.

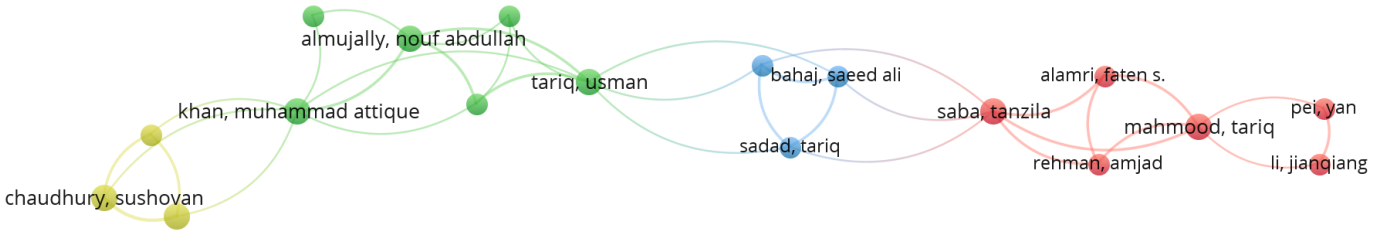


Fig. 8. Co-authorship analysis with authors - network visualization.

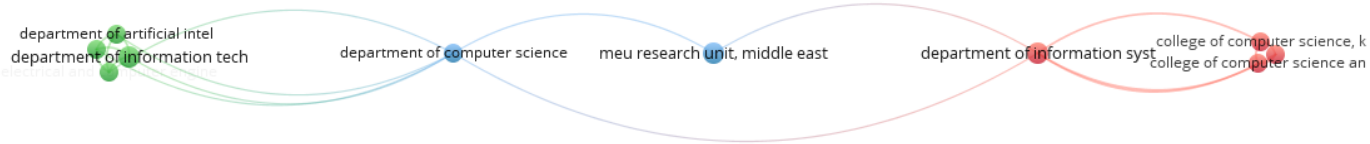


Fig. 9. Co-authorship analysis with organizations - network visualization.

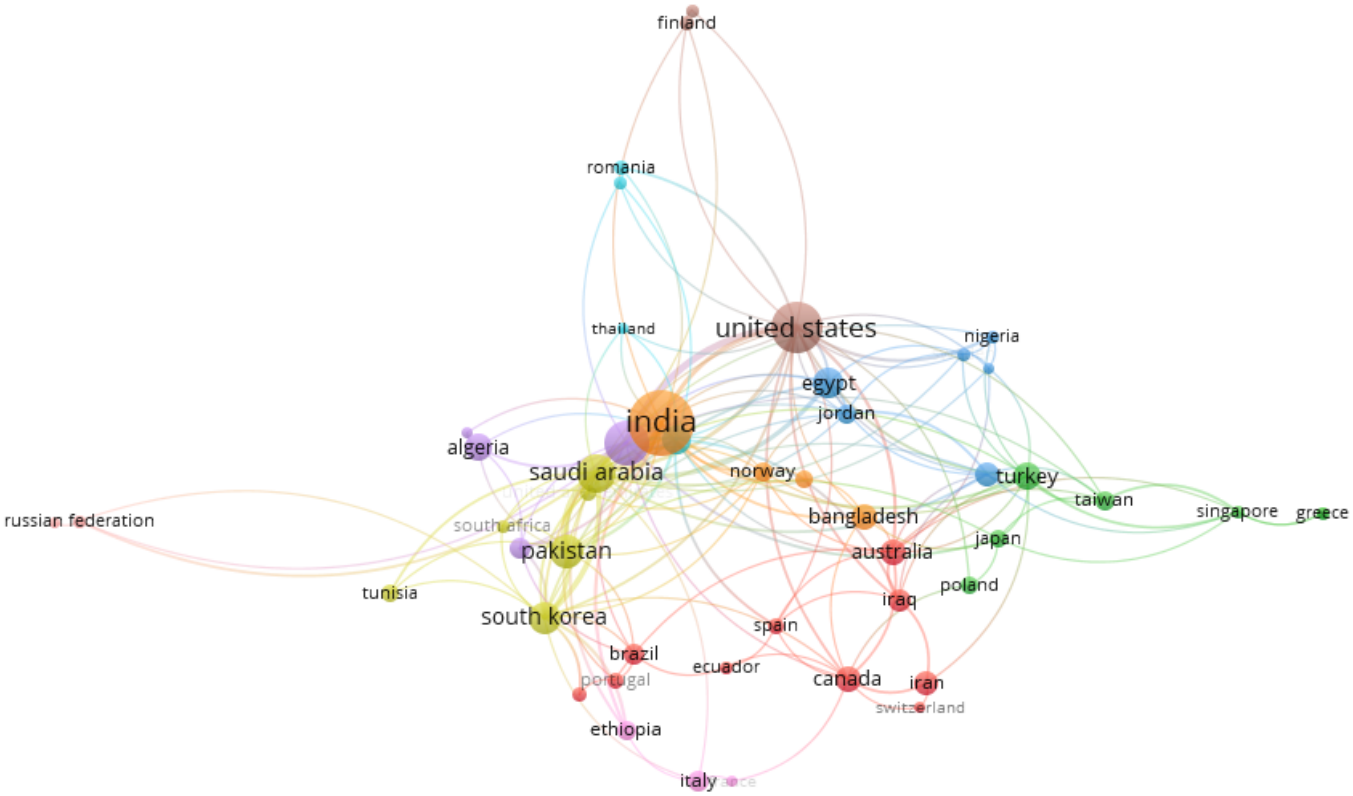


Fig. 10. Co-authorship analysis with countries - network visualization.

2) Co-occurrence analysis: Co-occurrence analysis has been conducted across various categories including all keywords, author keywords, and index keywords.

a) Co-occurrence with all Keywords: Out of the 2396 keywords identified in the selected publications, a total of 483 are analyzed, meeting the minimum occurrence criterion which is set to three. A network visualization is provided in Fig. 11.

b) Co-occurrence with author keywords: Examining the 779 author-assigned keywords, those with at least 3 occurrences are considered. A total of 111 entries met this threshold, as illustrated in Fig. 12.

c) Co-occurrence analysis based on index keywords: Out of the 2017 index keywords identified in the selected publications, 438 keywords met the minimum threshold of 3 occurrences. The network diagram in Fig. 13 reveals insights,

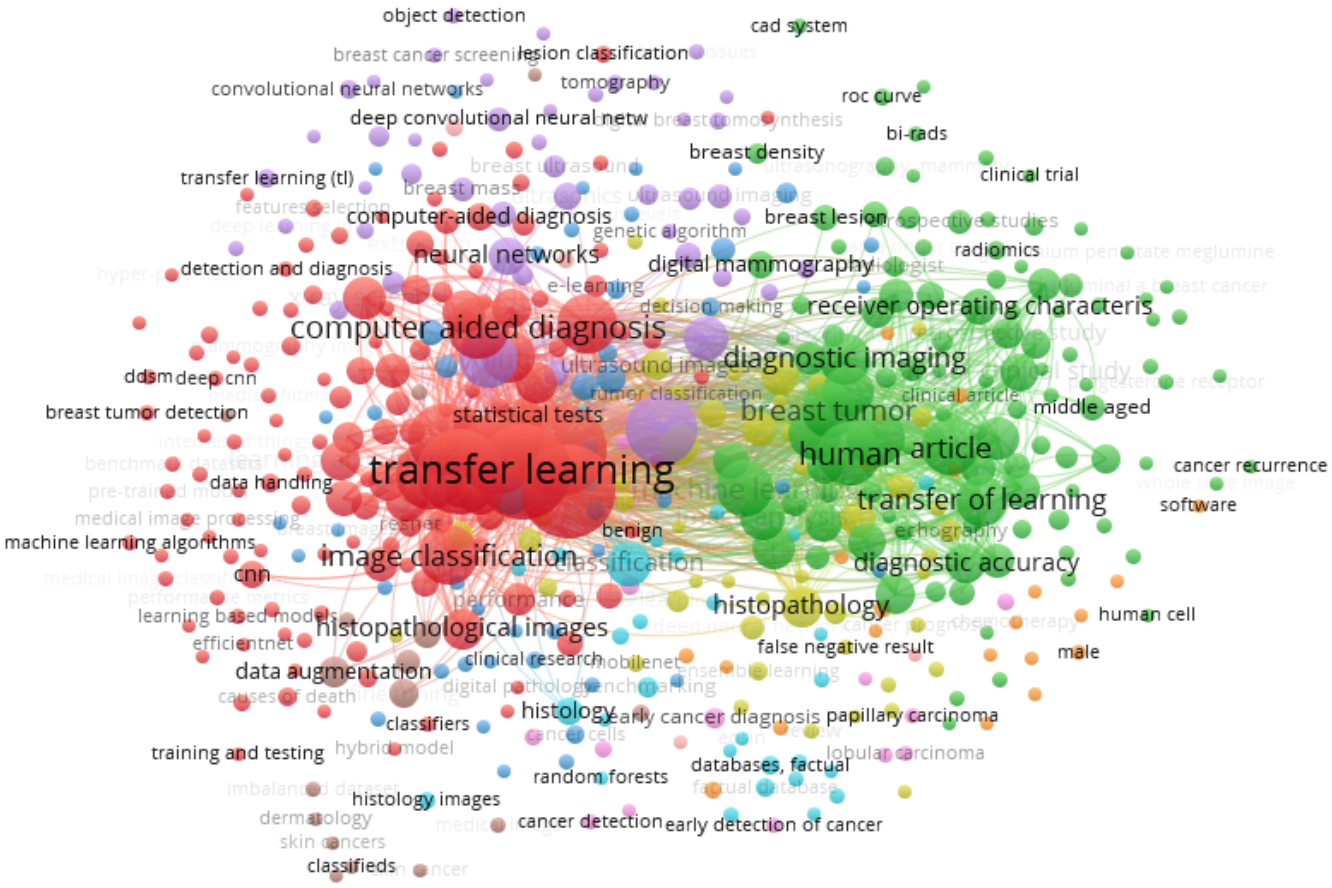


Fig. 11. Co-occurrence with all keywords - network visualization.

such as the keyword “Transfer Learning” having the highest total link strength of 4680, with 280 occurrences. Similarly, the keyword “Deep Learning” has 275 occurrences with a total link strength of 4719.

3) Citation analysis: This citation analysis examines various units of analysis, including documents, sources, authors, organizations, and countries.

a) Document citation analysis: Out of 452 publications, 222 documents meet the threshold of a minimum of 5 citations per document. Notably, Yap holds the top position with 588 citations. Some items in the network are disconnected. A total of 83 items are connected, which can be seen in Fig. 14.

b) Source citation analysis: The documents originate from 258 different sources, out of which 16 meet the threshold criteria of 5. Medical Physics and IEEE Journal of Biomedical and Health Informatics lead the list with 1178 and 678 citations, respectively, as shown in Fig. 15.

c) Author citation analysis: Among the 452 documents, 1680 authors have contributed. Considering a minimum of two documents per author, 215 authors are identified. Notably, Marit et al. has the highest citations (632) with 3 documents. Fig. 16 depicts the visualization of author citation analysis.

d) Organization citation analysis: Contributions from 1153 organizations are observed, with 10 organizations meeting the minimum threshold of 3 documents. The Department of IT/medical IT convergence engineering, Kumoh National Institute of Technology, Gumi, appears with 6 documents and 186 citations, as shown in Fig. 18.

e) Country citation analysis: The analyzed documents originate from 73 countries, with India, United States, and China ranking as the top three with 132, 73, and 51 documents, respectively. Their citation counts are 1496, 3491, and 1069, respectively. Fig. 17 depicts the visualization of country citation analysis.

The findings of this bibliometric analysis highlights a significant upward trajectory in research activities related to transfer learning for breast cancer detection over the past nine years. This growth highlights the increasing recognition and exploration of transfer learning as a viable solution to the challenges inherent in breast cancer detection, particularly those related to data limitations and classification accuracy. The dramatic rise from a mere three publications in 2016 to a peak of 133 in 2023 reflects a growing interest and investment in this area by the scientific community, indicating that transfer learning has become a focal point of innovation in medical imaging and oncology.

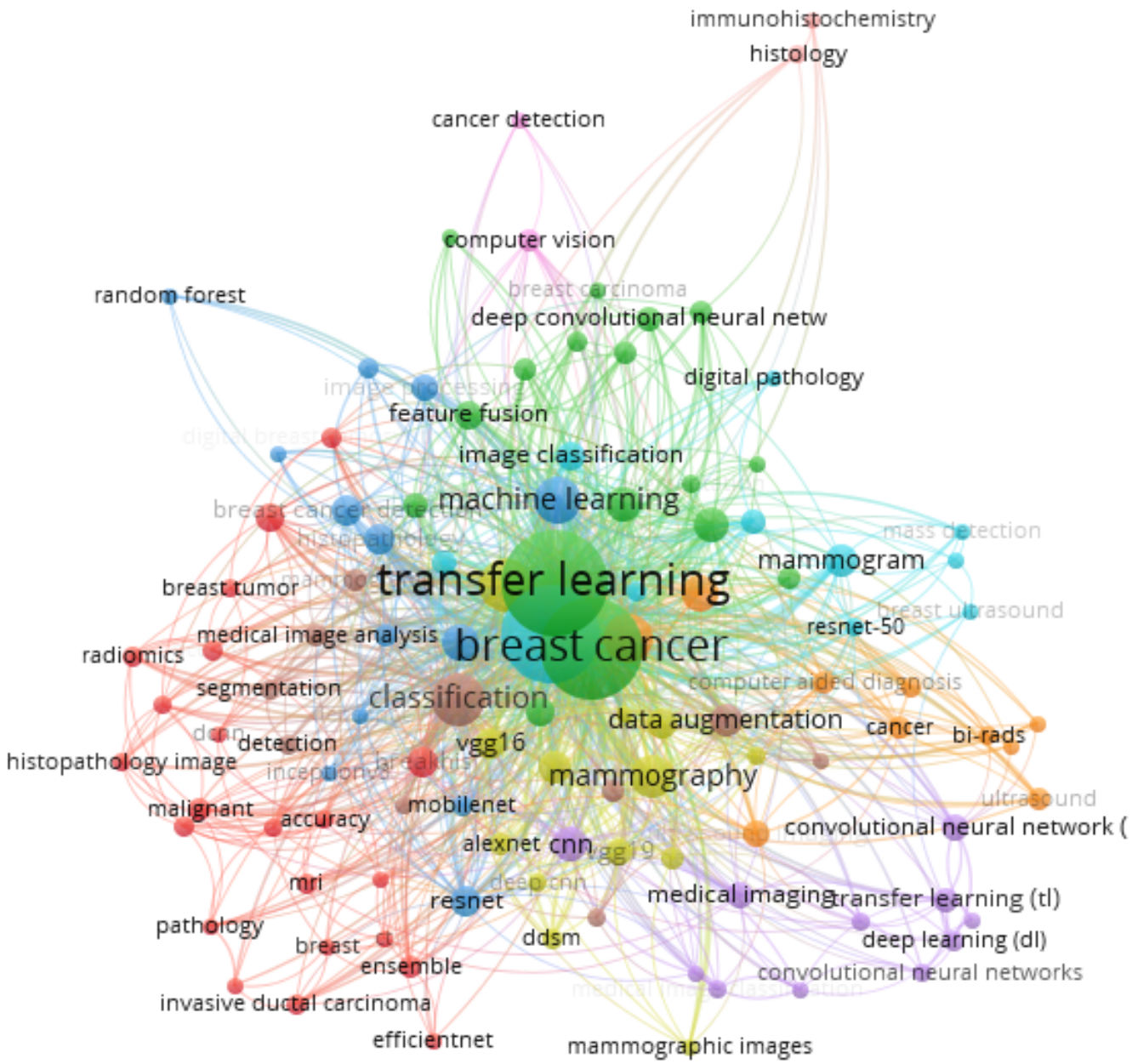


Fig. 12. Co-occurrence with author keywords - network visualization.

Moreover, the analysis identifies key contributors and institutions that are driving this research forward. Notably, Maryellen L. Giger emerged as the most prolific author, suggesting her significant influence and leadership in the field. Similarly, Princess Nourah bint Abdulrahman University is highlighted as a leading institution, demonstrating substantial involvement in advancing breast cancer detection technologies. The co-occurrence analysis of keywords further reveals critical research themes with terms like “Transfer Learning”, “Deep Learning”, and “Breast Cancer Detection” frequently appearing together. This suggests that these concepts are central to current research efforts and are likely to continue guiding future studies.

Overall, the insights gained from this bibliometric analysis provide a comprehensive overview of the current research landscape and identify potential areas for future exploration. The growing body of literature and the increasing collaboration among researchers and institutions underscore the promising potential of transfer learning techniques in revolutionizing breast cancer detection.

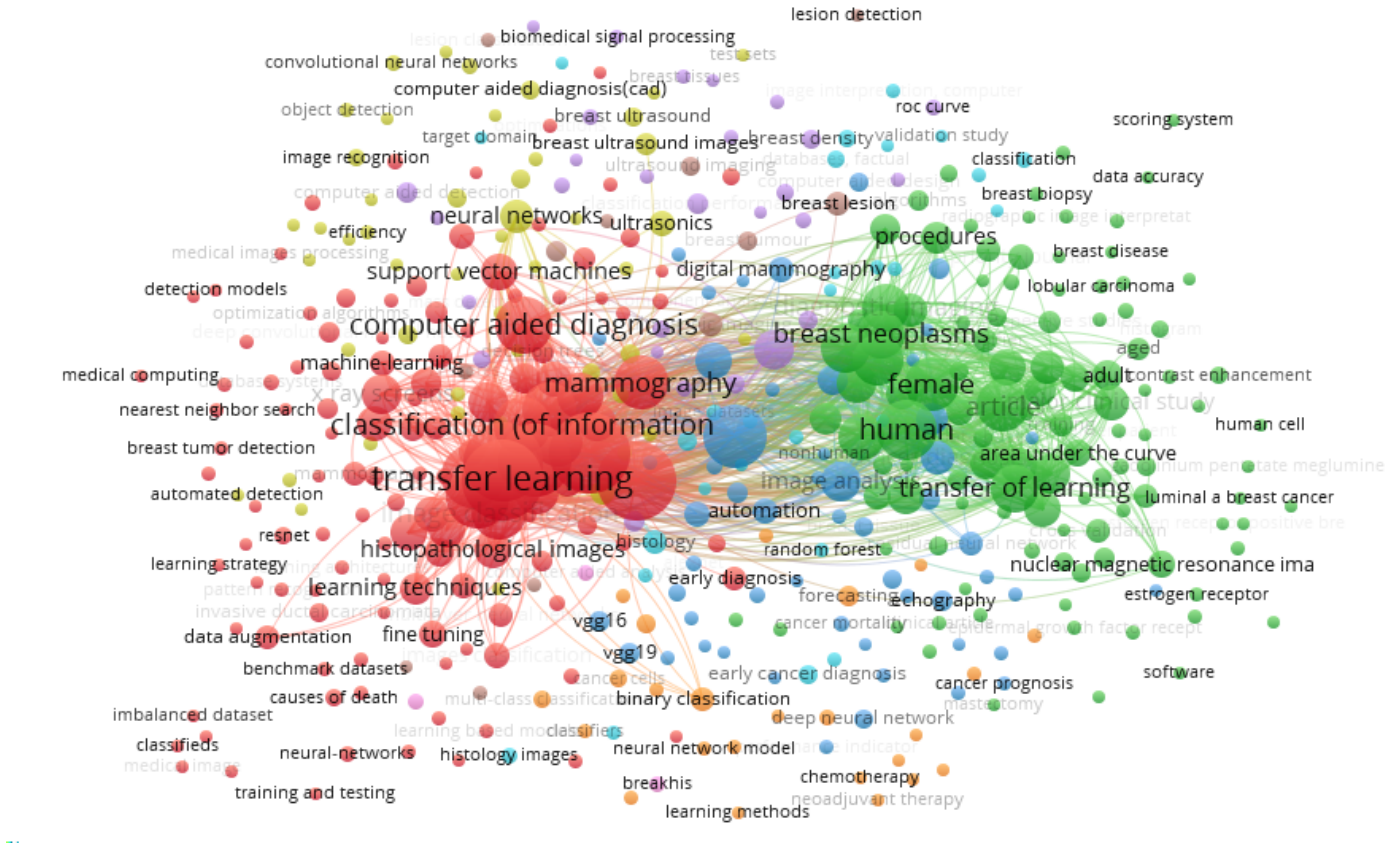


Fig. 13. Co-occurrence based on index keywords - network visualization.

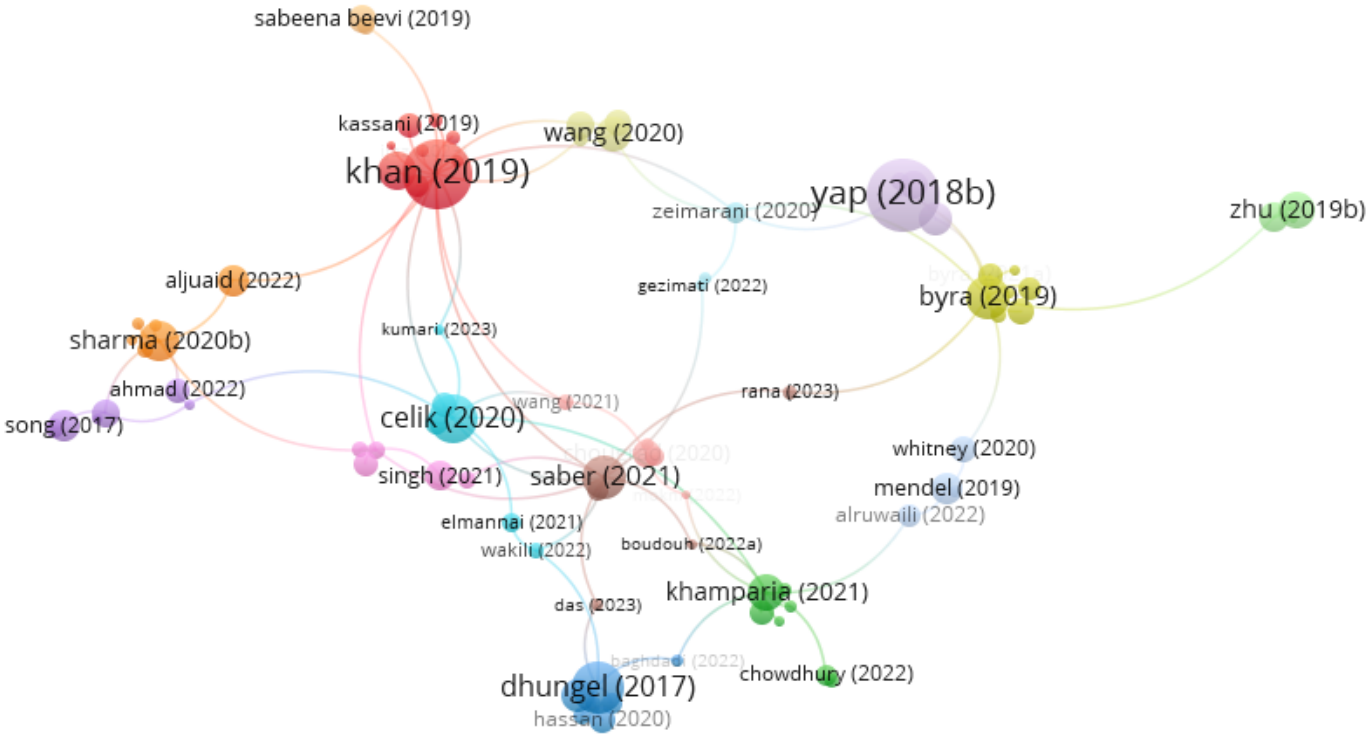


Fig. 14. Document citation analysis – network visualization.

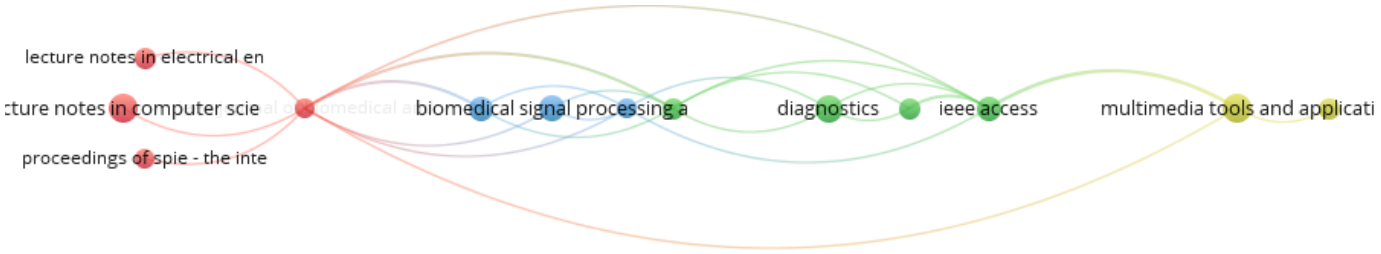


Fig. 15. Source citation analysis – network visualization.

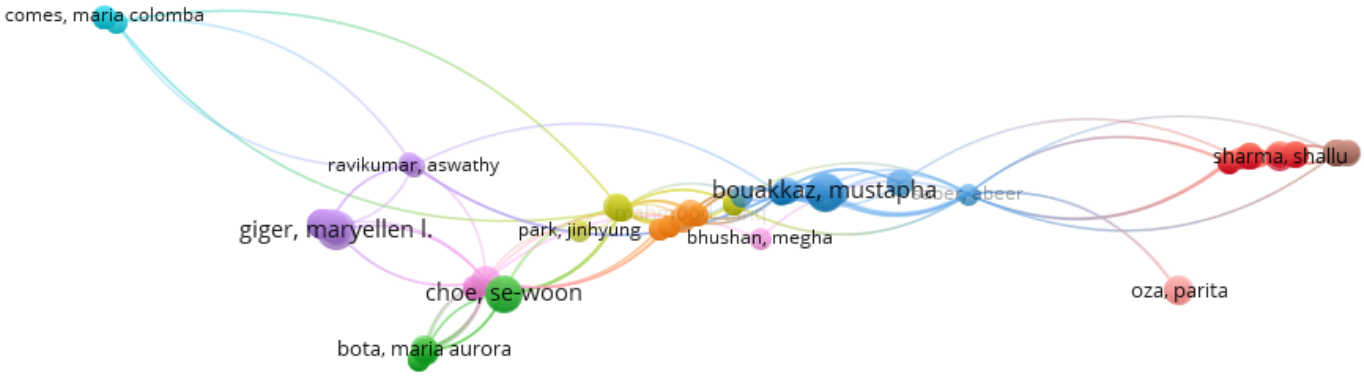


Fig. 16. Author citation analysis – network visualization.

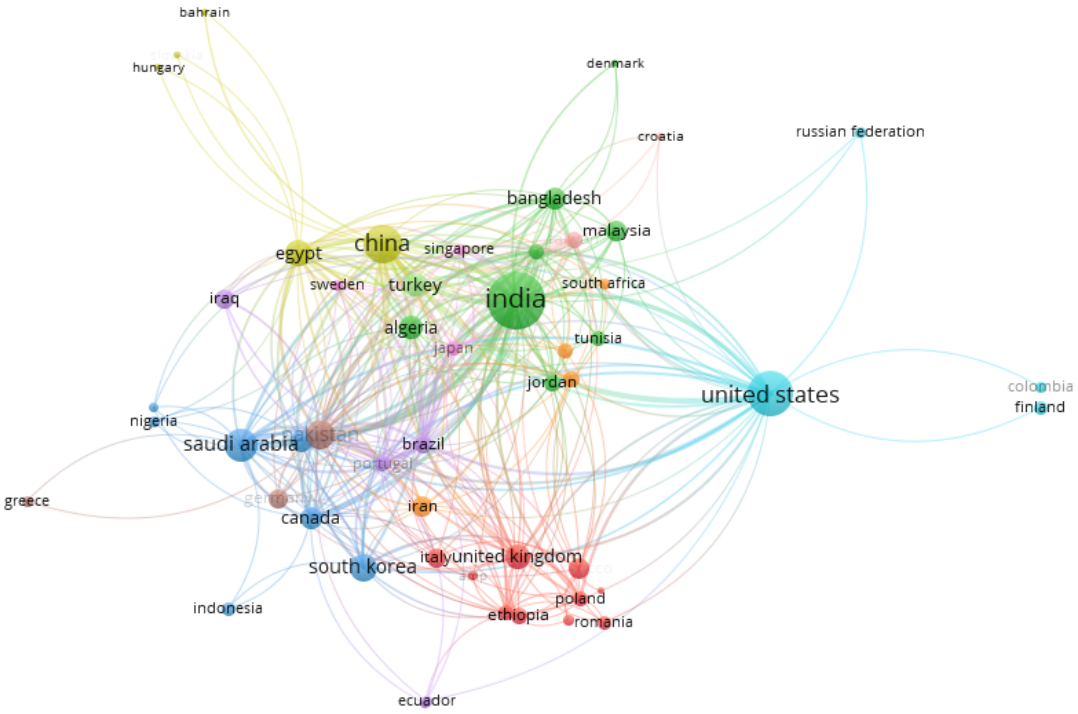


Fig. 17. Country citation analysis – network visualization.

V. Limitations

This study is limited by the exclusive use of the Scopus database for data collection, which may not capture all relevant publications on transfer learning in breast cancer detection. Additionally, the data is retrieved on April 21, 2024, and any documents published after this date are not included in the

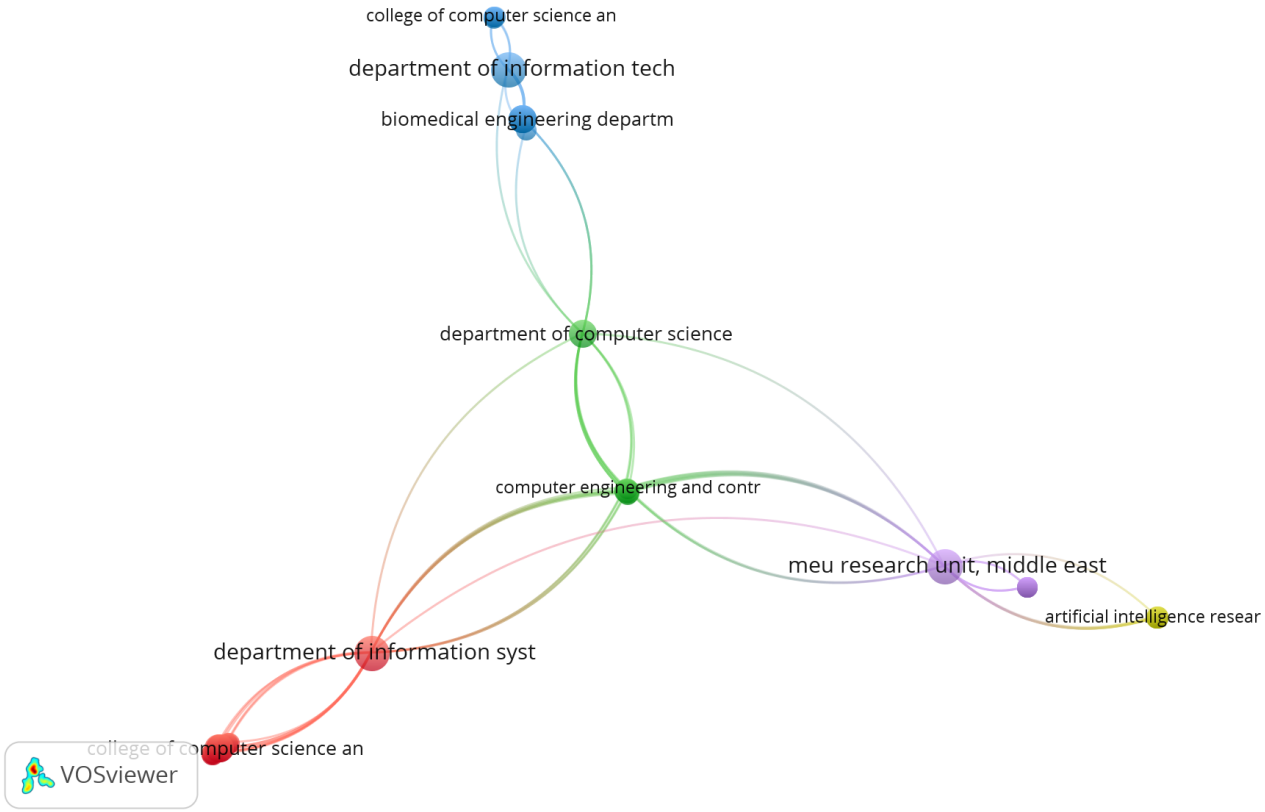


Fig. 18. Organization citation analysis – network visualization.

analysis. These limitations suggest that the findings might not fully represent the entire scope of research in this domain. A more comprehensive approach that includes other databases such as Web of Science, PubMed, and Google Scholar could provide a more holistic view of the research landscape.

VI. Conclusion and Future Directions

This bibliometric analysis reveals a significant upward trend in publications on transfer learning for breast cancer detection from 2016 to 2024, highlighting increasing research interest and advancements in the field. The results indicate that the volume of research on transfer learning for breast cancer detection has been steadily growing. In 2016, only 3 publications were available, but by 2023, this number had reached 133 publications, showing an upward trend in research interest. As of 2024, the results indicate continued activity in the field. The collaboration patterns in this research area reveal a mix of individual and institutional collaborations. Out of 1680 authors, 215 authors have published at least two papers, with the largest connected set comprising 18 authors. Princess Nourah bint Abdulrahman University has emerged as a leading institution, while China and India have shown significant research contributions, both through single-country and multi-country publications.

Future research will consider conducting a systematic review to complement this bibliometric analysis, providing deeper insights into the qualitative aspects of the studies, and identifying gaps in the existing literature. Additionally, comparative studies examining the research output on breast cancer detection using transfer learning across different countries, especially in developing regions, would be beneficial. Expanding the scope to include multiple databases and exploring collaborative patterns in diverse geographical regions could improve understanding of global research trends and foster international collaboration. These steps will help in advancing the field further and potentially lead to breakthroughs in early diagnosis and improved patient outcomes in breast cancer detection.

Data Availability

Data are available on request.

Statements and Declarations

All authors have no conflict of interest with anyone.

References

[1] M. Nasser and U. K. Yusof, "Deep learning based methods for breast cancer diagnosis: A systematic review and future direction," *Diagnostics*, vol. 13, no. 1, p. 161, 2023.

[2] S. M. McKinney, M. Sieniek, V. Godbole, J. Godwin, N. Antropova, H. Ashrafian, and S. Shetty, "International evaluation of an ai system for breast cancer screening," *Nature*, vol. 577, no. 7788, pp. 89–94, 2020.

[3] K. C. Oeffinger, E. T. H. Fontham, R. Etzioni, A. Herzig, J. S. Michaelson, and Y.-C. T. Shih, "Breast cancer screening for women at average risk," *JAMA*, vol. 314, p. 1599, 2015.

[4] A. Saber, M. Sakr, O. M. Abo-Seida, A. Keshk, and H. Chen, "A novel deep-learning model for automatic detection and classification of breast cancer using the transfer-learning technique," *IEEE Access*, vol. 9, pp. 71 194–71 209, 2021.

[5] H. E. Kim, A. Cosa-Linan, N. Santhanam, M. Jannesari, M. E. Maros, and T. Ganslandt, "Transfer learning for medical image classification: a literature review," *BMC Medical Imaging*, vol. 22, no. 1, p. 69, 2022.

[6] S. Ahmad, S. Ur Rehman, A. Iqbal, R. K. Farooq, A. Shahid, and M. I. Ullah, "Breast cancer research in pakistan: A bibliometric analysis," *SAGE Open*, vol. 11, no. 3, p. 21582440211046934, 2021.

[7] A. H. Syed and T. Khan, "Evolution of research trends in artificial intelligence for breast cancer diagnosis and prognosis over the past two decades: A bibliometric analysis," *Frontiers in Oncology*, vol. 12, p. 854927, 2022.

[8] S. S. M. Khairi, M. A. A. Bakar, M. A. Alias, S. A. Bakar, C. Y. Liong, N. Rosli, and M. Farid, "Deep learning on histopathology images for breast cancer classification: A bibliometric analysis," in *Healthcare*, vol. 10, no. 1. MDPI, December 2021, p. 10.

[9] W. Zhou, Z. Deng, Y. Liu, H. Shen, H. Deng, and H. Xiao, "Global research trends of artificial intelligence on histopathological images: a 20-year bibliometric analysis," *International Journal of Environmental Research and Public Health*, vol. 19, no. 18, p. 11597, 2022.

[10] S. A. Joshi, A. M. Bongale, and A. Bongale, "Breast cancer detection from histopathology images using machine learning techniques: a bibliometric analysis," *Library Philosophy and Practice*, vol. 5376, pp. 1–29, 2021.

[11] i. Ozen Cinar, "Bibliometric analysis of breast cancer research in the period 2009–2018," *International Journal of Nursing Practice*, vol. 26, no. 3, p. e12845, 2020.

[12] M. Tober, "Pubmed, sciencedirect, scopus or google scholar–which is the best search engine for an effective literature research in laser medicine?" *Medical Laser Application*, vol. 26, no. 3, pp. 139–144, 2011.

An Adaptive Levy Flight Chicken Swarm Optimization with Differential Evolution for Function Optimization Problem

WEN-JUN LIU[1], AZLAN MOHD ZAIN[2], MOHAMAD SHUKOR BIN TALIB[3], SHENG-JUN MA[4]
Faculty of Computing, Universiti Teknologi Malaysia, Skudai, 80310, Malaysia[1,2,3]
Electrical and Electronic Engineering Department, Auckland University of Technology, Auckland, New Zealand[4]

***Abstract*—This study proposes an improved swarm algorithm, Adaptive Levy Flight Chicken Swarm Optimization with Differential Evolution (ALCSODE), to overcome the low convergence accuracy and imbalance between exploration and exploitation in the original CSO algorithm. The method incorporates adaptive perturbation based on individual differences and a differential evolution mechanism into the rooster update process. An elitism preservation strategy is also applied to enhance population stability and information sharing. The algorithm is evaluated on 24 benchmark functions, including unimodal, high-dimensional multimodal, and CEC2022 functions. Performance metrics such as search trajectories and convergence curves are used to assess its effectiveness. Experimental results show that ALCSODE achieves a better exploration–exploitation trade-off and shows statistically superior performance over seven classical algorithms, confirming its potential as an effective tool for solving complex optimization problems.**

Keywords*—Chicken swarm optimization; levy flight; differential evolution algorithm; adaptive adjustment strategy; function optimization

I. Introduction

As the dimensionality and complexity of real-world problems increase, optimization tasks exhibit diverse characteristics, such as the coexistence of discrete and continuous variables, low- and high-dimensional settings, and both convex and non-convex landscapes [1]. These challenges significantly limit the effectiveness of traditional analytical and gradient-based methods. In non-smooth, non-convex, or high-dimensional search spaces, gradients are often difficult to compute, and numerical methods are prone to local optima with slow convergence, which severely hampers solution quality and efficiency.

To overcome these limitations, metaheuristic algorithms have emerged as a powerful alternative. Inspired by natural processes such as evolution and swarm intelligence, these algorithms iteratively refine candidate solutions through stochastic operations, offering derivative-free optimization and strong adaptability [2]. Their core strength lies in the ability to balance global exploration and local exploitation [3]: exploration aims to identify promising regions in the search space, while exploitation focuses on refining existing solutions toward the global optimum. The balance between exploration and exploitation is crucial for achieving good optimization performance. Excessive exploration may reduce convergence speed, while excessive exploitation can lead to premature convergence. Therefore, designing algorithms that can effectively coordinate both aspects remains a central goal in metaheuristic development. Generally, metaheuristic algorithms are categorized into four major types: evolution-based, physics-based, mathematics-based, and swarm intelligence-based methods.

The first category comprises general population-based metaheuristic algorithms, which are inspired by biological evolution and natural selection. These algorithms simulate the competitive survival process within a population to iteratively improve solution quality. Representative methods include Genetic Algorithms (GA) [4], Differential Evolution (DE) [5], and Evolution Strategies (ES) [6]. They typically employ operators such as selection, crossover, and mutation to evolve the population, ensuring both diversity maintenance and convergence toward superior solutions.

The second category consists of physics-inspired optimization algorithms, which guide the search process by mimicking natural physical phenomena such as thermodynamics, gravitational force, and electromagnetic interactions. For instance, Simulated Annealing (SA) [7] is based on the annealing process of materials and helps the algorithm escape local optima. The Gravitational Search Algorithm (GSA) [8] and the Black Hole Algorithm (BH) [9] leverage mechanical or astrophysical models to drive the search process. These methods generally offer strong global search capabilities and are well suited to high-dimensional and complex optimization problems. Moreover, Harmony Search (HS) [10], inspired by musical improvisation, seeks optimal solutions. Dynamic Dimension Adjustment HS (DDA-HS) [11] adapts the search space and employs an absorbing Markov chain to improve exploration–exploitation balance, enhancing fuzzy rule extraction in neural networks for better classification and interpretability.

The third category includes mathematically inspired algorithms, which regulate the search process through rigorous mathematical strategies. The INFO [12] adopts an averaging-based update mechanism and a vector combination strategy to enhance convergence speed and local exploitation performance, while alleviating premature convergence and population stagnation. Similarly, the SHIO [13] incorporates a guidance mechanism based on the average of historical best solutions to balance exploration and exploitation, thus improving global performance and avoiding local optima.

The fourth category, swarm intelligence-based optimization algorithms, is inspired by collective foraging, migration, and cooperative decision-making behaviors observed in biological systems. These algorithms achieve global optimization through simple interactions among individuals and local adap-

tive search without centralized control, exhibiting strong self-organization, robustness, and parallelism [14].

In recent years, several novel algorithms have emerged in this domain. The Snow Geese Algorithm (SGA) [15] models global guidance and shortest path search by mimicking the V-formation and energy-sharing strategies of migratory geese. The Sled Dog Optimizer (SDO) [16] simulates sled dog team formation and dynamic role-switching to construct a multi-stage cooperative optimization mechanism. The HawkFish Optimization Algorithm (HFOA) [17] incorporates dual-fitness and vision-based dynamic strategies inspired by sex-switching behavior, demonstrating both stability and efficiency in solving multimodal problems. Classical swarm-based algorithms, including Particle Swarm Optimization (PSO) [18], Ant Colony Optimization (ACO) [19], Artificial Bee Colony (ABC) [20], Artificial Fish Swarm Algorithm (AFSA) [21], and Bat Algorithm (BA) [22], along with emerging variants such as Cuckoo Search (CS) [23], Hybrid Harmony Search with Cuckoo Search (HS-CS) [24], Krill Herd Optimization [25], Chaotic Fruit Fly Optimization Algorithm [26], and Social Spider Optimization [27], have found wide applications in fields such as computer science [28], engineering design [29], machine learning [30], image processing [31], and NP-complete problem solving [32].

The Chicken Swarm Optimization (CSO) algorithm [33] is a swarm intelligence technique inspired by the hierarchical structure and foraging behavior of chicken flocks. It assigns individuals as roosters, hens, or chicks, each applying distinct search strategies to achieve cooperative optimization and improve its capability to locate the global optimum [34].

However, the original CSO algorithm faces several limitations, such as a tendency to fall into local optima, slow convergence speed, and high sensitivity to parameter settings. To address these issues, various improved variants have been proposed. To strengthen global exploration, Huang et al. [35] introduce a global best-guided strategy combined with dynamic inertia weight, significantly improving the search capability of the swarm. To mitigate premature convergence, Kong et al. [36] propose a dual-guidance mechanism in which chicks learn from both hens and roosters, thereby enhancing their ability to escape local optima. Shi and Gao [37] integrate CSO with the Artificial Bee Colony (ABC) algorithm to leverage ABC's global search ability for better exploration-exploitation balance. A hybrid strategy is also developed by combining CSO with Particle Swarm Optimization (PSO) to enhance information exchange and population diversity [38]. Zhang et al. [39] further propose L-QCSO by incorporating Levy flight and quantum behavior, aiming to balance local exploitation and global exploration while improving convergence accuracy. Lin et al. [40] enhance the robustness and precision of CSO by modifying the position update rules for roosters and hens. Theoretical advancements have also been made, such as models based on Markov chains and convergence analysis, which provide a reliable foundation for parameter tuning and structural design [41]. In practical applications, CSO is often combined with other optimization or learning models. For example, Afzal et al. [42] present CSO-OEL by integrating CSO with the Optimized Extreme Learning Machine, resulting in improved convergence speed, stability, and overall efficiency.

Despite the significant achievements of various optimization algorithms in solving complex problems, the "No Free Lunch" theorem [43] states that no single algorithm can perform optimally across all problem domains. The performance of an algorithm is generally problem-dependent, making the NFL theorem a fundamental driving force behind the development of problem-specific optimization strategies.

The Chicken Swarm Optimization (CSO) algorithm has attracted attention for its simplicity and effective modeling of social behavior. However, existing studies reveal persistent limitations, including weak role coordination, slow convergence in complex landscapes, and sensitivity to parameter settings—often caused by the imbalance between exploration and exploitation. These issues hinder the algorithm's scalability, adaptability, and solution quality in high-dimensional or dynamic problems. To improve convergence accuracy and address these limitations, this study draws on recent advances in swarm intelligence and proposes that a high-performance optimizer should integrate multi-strategy cooperation and adaptive control mechanisms.

To enhance the scalability, robustness, and convergence behavior of the traditional Chicken Swarm Optimization (CSO), this research proposes a novel algorithm, Adaptive Levy Flight Chicken Swarm Optimization with Differential Evolution (ALCSODE). The main innovation of ALCSODE lies in its strategic integration of multiple adaptive and cooperative mechanisms, each designed to explicitly tackle key structural deficiencies of CSO, rather than simply stacking heuristic components. The contribution is a modular, coordinated optimization framework that strengthens global search capabilities, improves role collaboration, and enhances adaptability to complex landscapes.

Specifically, in the rooster phase, an adaptive differential perturbation mechanism enhances global exploration through individual difference vectors, supported by a fitness-based step-size strategy that dynamically adjusts the search range to balance exploration and exploitation. A non-inertia weight mechanism further improves convergence stability by adapting parameter intensity to the current search stage. In the hen phase, a Levy flight strategy utilizes long-tailed jumps to escape local optima and accelerate convergence. Additionally, an elitism preservation and information-sharing mechanism maintains diversity and guides inferior individuals toward high-quality solutions, improving convergence without compromising population diversity. The scientific contributions of this research are summarized as follows:

- The novelty of this study lies in the design of a new optimizer, ALCSODE, which incorporates an adaptive differential perturbation mechanism, Levy flight strategy, non-inertia weight adjustment, and elitism preservation into the Chicken Swarm Optimization (CSO) framework.
- The proposed ALCSODE algorithm addresses key limitations of the original CSO, including weak role coordination, premature convergence, and sensitivity to parameter settings caused by the imbalance between exploration and exploitation.
- A modular, adaptive optimization framework is developed, with each component designed to enhance

global exploration, convergence stability, and role collaboration in complex search spaces.

- The algorithm's core strategies are formally described and mathematically modeled to ensure transparency and reproducibility.
- ALCSODE is evaluated on 24 benchmark functions, covering unimodal, multimodal, and the CEC2022 test suite, to verify its scalability and effectiveness in diverse optimization scenarios.
- Comparative experiments with seven established metaheuristic algorithms show that ALCSODE consistently achieves superior performance and robustness in most test cases.

The remainder of this study is organized as follows: Section II reviews the standard CSO and DE algorithms; Section III details the implementation steps of ALCSODE; Section IV presents the simulation experiments and analysis; and Section V summarizes the whole manuscript and outlines future research directions.

II. Related Work and Background Algorithms

Several swarm intelligence algorithms have been proposed in recent years, among which the Chicken Swarm Optimization (CSO) and the Differential Evolution (DE) algorithms are widely used due to their strong optimization capability. This section briefly introduces the basic principles of the standard CSO and DE algorithms, which form the foundation of our proposed ALCSODE method.

A. The Chicken Swarm Optimization

Swarm intelligence algorithms have demonstrated remarkable advantages in solving complex optimization problems. The Chicken Swarm Optimization (CSO) algorithm achieves efficient search performance by simulating the hierarchical behavior of chicken groups [34]. In this algorithm, the population is categorized into three distinct roles: roosters (leaders), hens, and chicks, which correspond to exploration, cooperation, and following behaviors, respectively.Typically, individuals with the best fitness values are assigned as roosters, while those with the worst fitness values are designated as chicks. The remaining individuals act as hens. The entire chicken swarm is further divided into several subgroups, each led by one rooster accompanied by a number of hens and chicks. To enhance the algorithm's adaptability in solving complex multimodal problems, the identities of individuals and the parent–child relationships are periodically redefined.

Let N be the total number of chickens, and R_N, H_N, C_N, and M_N denote the number of roosters, hens, chicks, and mother hens, respectively. The search space is defined as $\mathbb{R}^D$. The position of the i-th individual in the j-th dimension at iteration t is denoted by $x_{i,j}^t$, where $i \in [1, \ldots, N]$, $j \in [1, \ldots, D]$, and t is the current iteration. The position of roosters is updated using a random walk and competition mechanism, formulated as follows [see Eq. (1) and Eq. (2)]:

$$x_{i,j}^{t+1} = x_{i,j}^t(1 + \mathcal{N}(0, \sigma^2)), \tag{1}$$

$$\sigma^2 = \begin{cases} \exp\left(\frac{f_k - f_i}{|f_i| + \epsilon}\right) & \text{if } f_k > f_i, \\ 1 & \text{otherwise,} \end{cases} \tag{2}$$

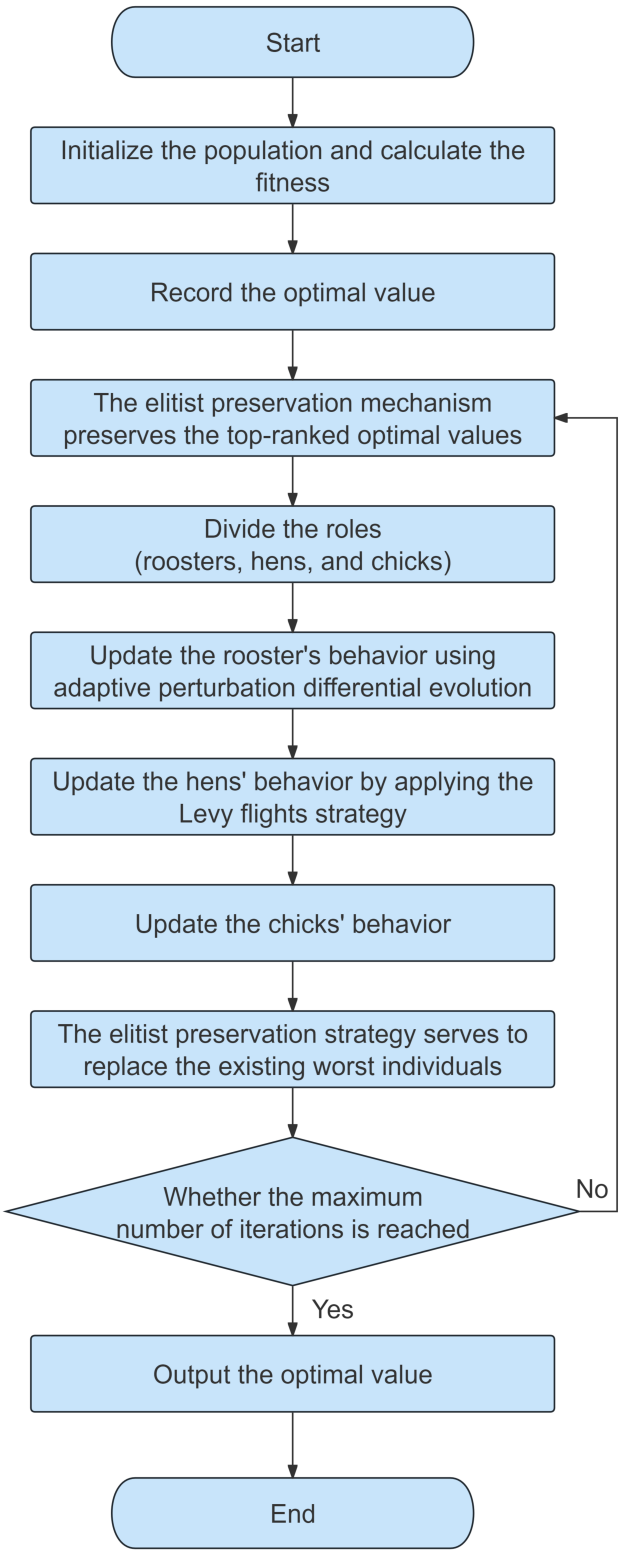


Fig. 1. The flowchart of ALCSODE algorithm.

where, f_i represents the fitness of the current rooster,f_k is the fitness of a randomly selected rooster, andϵ is a small constant used to avoid division by zero.

For hens, their position update combines leadership from the roosters and social learning. The corresponding update equation is defined as follows [see Eq. (3)]:

$$x_{i,j}^{t+1} = x_{i,j}^{t} + c_1 r_1 (x_{r,j}^{t} - x_{i,j}^{t}) + c_2 r_2 (x_{m,j}^{t} - x_{i,j}^{t}), \quad (3)$$

where, $x_{r,j}^{t}$ denotes the position of the leading rooster, $x_{m,j}^{t}$ represents the position of another randomly selected hen, c_1 and c_2 are learning factors (typically defined as $c_1 = \exp(f_i/f_r)$, $c_2 = \exp(f_m - f_i)$), and $r_1, r_2 \in [0, 1]$ are random numbers.

As for the chicks, they strictly follow the movement of the mother hens. The update equation is as follows [see Eq. (4)]:

$$x_{i,j}^{t+1} = x_{i,j}^{t} + F(x_{h,j}^{t} - x_{i,j}^{t}). \quad (4)$$

where, $x_{h,j}^{t}$ denotes the position of the hen, and $F \in (0, 2]$ is the following step size factor.

B. The Differential Evolution Algorithm

Differential Evolution (DE [44]) is a population-based stochastic optimization algorithm proposed by Storn and Price in 1997. It is an important branch of evolutionary computation. The core idea of DE is to simulate the mutation, crossover, and selection operations in biological evolution to efficiently search for the global optimal solution in continuous space. The DE algorithm begins by initializing a fixed-size population, with each individual representing a candidate solution in the solution space (referred to as a target vector). The optimization process then iteratively refines the solution, gradually approaching the optimal one. During the mutation step, the algorithm generates mutation vectors using differential operations. Common strategies include generating mutation vectors by linearly combining three random individuals (DE/rand/1) and introducing the global best solution to guide mutation (DE/rand-to-best/1), which helps overcome the limitations of traditional mutation methods in genetic algorithms (GA [4]). In the crossover phase, the mutation vectors and target vectors are probabilistically mixed to create trial vectors, enhancing population diversity. Finally, a greedy selection strategy is used to retain individuals with better fitness for the next generation, continuously improving the solution quality. The DE algorithm is widely applied in single-objective and multi-objective optimization, combinatorial optimization, and engineering problems due to its simple structure, few parameters, and high efficiency.

III. The Proposed ALCSODE Method

In this section, we present our Adaptive Levy Flight Chicken Swarm Optimization with Differential Evolution algorithm, termed ALCSODE, whose detailed pseudocode is provided in Algorithm 1 and overall workflow is illustrated in Fig. 1. For clarity and focus, our method follows the standard CSO framework for initialization and role assignment.

The algorithm is divided into four core components, highlighting two innovative improvements: 1) The adaptive perturbation-based differential evolution strategy for rooster behavior updating. 2) The elitism preservation mechanism to improve global robustness.

A. Adaptive Perturbation-Based Differential Evolution for Rooster Behavior Updating

In Chicken Swarm Optimization (CSO) algorithms, rooster behavior plays a pivotal role in guiding population exploration and maintaining swarm diversity, as documented in foundational studies [45]. However, traditional updating mechanisms exhibit significant limitations: 1) single-strategy search susceptibility to local optima causing premature convergence, 2) insufficient spatial perturbation, and 3) fixed parameters lacking environmental adaptability [46]. To address these issues, our study proposes a method integrating individual adaptability perturbation with the differential evolution algorithm [5]. We use the differential information among individuals to guide the search to high-quality regions. The adaptive perturbation dynamically adjusts the search step size based on individual fitness. This synergy balances global exploration and local exploitation, avoiding search imbalance from single-strategy approaches. Additionally, parameters can adaptively adjust their intensity, enabling effective regulation of search efforts at different stages. The detailed method is introduced in the following section.

1) Calculation of adaptive differential mutation vector: To enhance global exploration in early stages and intensify local exploitation in later phases, this study introduces an adaptive update strategy for the scaling factor F and crossover probability CR in Differential Evolution. Let t denote the current iteration and T the maximum iterations, with initial values F_0 and CR_0. The adaptive update equations are defined as Eq. (5) and Eq. (6):

$$F_t = F_0 \left(1 - \frac{t}{T}\right)^{\theta} + F_{min}, \quad (5)$$

$$CR_t = CR_0 \left(1 - \frac{t}{T}\right)^{\theta} + CR_{min}, \quad (6)$$

where, CR_{min} and F_{min} are both set to 0.1, and $\theta > 0$ controls the decay rate. This adaptive mechanism ensures larger step sizes and higher mutation probabilities during initial phases, facilitating escape from local optima, while progressively decreasing parameters promote stable convergence as iterations advance.

Using differential evolution, our method leverage information from other population individuals to guide global search directions. Three distinct individuals are randomly selected from the population, denoted as $\mathbf{x}_{r_1}$, $\mathbf{x}_{r_2}$, $\mathbf{x}_{r_3}$. The differential evolution mutation vector is constructed as follows [see Eq. (7)]:

$$\mathbf{d}_i^{t+1} = \mathbf{x}_{r_1}^{t} + F_t \left(\mathbf{x}_{r_2}^{t} - \mathbf{x}_{r_3}^{t}\right). \quad (7)$$

where, t denotes the iteration, F_t represents the adaptive scaling factor. $\mathbf{d}_i^{t}$ provides a new search direction based

on population differential information. This step effectively utilizes inter-population differential information to introduce diversity in global search, facilitating exploration of potential optimal regions.

2) Calculation of individual-adaptive perturbation factor: Individual-specific perturbations are applied to candidate solutions (representing rooster behaviors in the swarm) to enhance search capabilities of inferior individuals. Let f_i denote the fitness of the i-th individual and f_b the best fitness in the population. The perturbation standard deviation σ_i is defined as Eq. (8):

$$\sigma_i^2 = \begin{cases} 1, & f_i \le f_b \\ \exp\left(\frac{f_b - f_i}{|f_i| + \varepsilon}\right), & f_i > f_b \end{cases} \tag{8}$$

where, ε represents a minimal positive constant to prevent division by zero. The perturbation step size δ_i is then sampled from a normal distribution with zero mean and variance σ_i^2, as in Eq. (9):

$$\delta_i \sim \mathcal{N}\big(0, \sigma_i^2\big). \tag{9}$$

Fitness-adaptive perturbation adjustment enables inferior individuals to receive larger perturbations, facilitating escape from local optima, while superior individuals maintain stability, ensuring convergence quality.

3) Updating the roosters behavior: Integrating global differential search information with current rooster states, the next generation is generated as follows. Let $\mathbf{x}_i^t$ denote the current individual position. With a trade-off parameter β, the next generation vector is defined as Eq. (10):

$$\mathbf{x}_i^{t+1} = \alpha\, \mathbf{d}_i^{t+1} + \delta_i\, \mathbf{x}_i^t, \tag{10}$$

where, $\alpha \in (0, \infty)$ controls the relative weight between global exploration and local exploitation. $\mathbf{d}_i^{t+1}$ represents the differential evolution mutation vector computed by Eq. (7), while δ_i denotes the adaptive perturbation factor calculated from Eq. (9).

To prevent excessive localization of new solutions, a component-wise crossover strategy is employed for dimension-by-dimension position updating. For each dimension j, the next generation rule is defined as Eq. (11):

$$x_{i,j}^{'t+1} = \begin{cases} x_{i,j}^{t+1}, & \text{if } rand(0,1) \le CR_t \text{ or } j = j_{rand} \\ x_{i,j}^{t}, & otherwise \end{cases} \tag{11}$$

where, CR_t denotes the crossover probability calculated by the Eq. (6), and j_{rand} ensures at least one dimension is updated.

By integrating the differential evolution mutation vector with individual-specific perturbations and updating the rooster vector through component-wise crossover, this strategy maintains population diversity while effectively balancing global exploration and local exploitation, thereby enhancing convergence speed and stability.

B. Levy Flight-Based Hen Behavior Optimization

To enable hen individuals to adaptively adjust search step sizes based on fitness information during updating, dynamic scaling factors are first constructed from fitness differences between the current individual and two randomly selected individuals. The scaling factors are defined as Eq. (12) and Eq. (13):

$$\lambda_1 = \exp\left(\frac{f_i - f_l}{|f_i| + \varepsilon}\right), \tag{12}$$

$$\lambda_2 = \exp\left(f_k - f_i\right), \tag{13}$$

where, f_i denotes the fitness of the current hen individual; f_l and f_k represent the fitness values of randomly selected individuals; ε is a small positive constant preventing division by zero; λ_1 and λ_2 reflect the relative fitness advantage or disadvantage of the current individual, providing dynamic adjustment factors for subsequent position updating.

To enhance global exploration capability, the Levy flight function is employed. Given the Levy parameter β, the scale parameter σ_u is first calculated as Eq. (14):

$$\sigma_u = \left[\frac{\Gamma(1+\beta)\, \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right)\, \beta\, 2^{\frac{\beta-1}{2}}}\right]^{\frac{1}{\beta}}, \tag{14}$$

where, $\Gamma(\cdot)$ denotes the gamma function. The Levy flight step size is calculated as Eq. (15):

$$L = \sigma_u \left(u^2 + v^2\right)^{\frac{1}{2\beta}}, \tag{15}$$

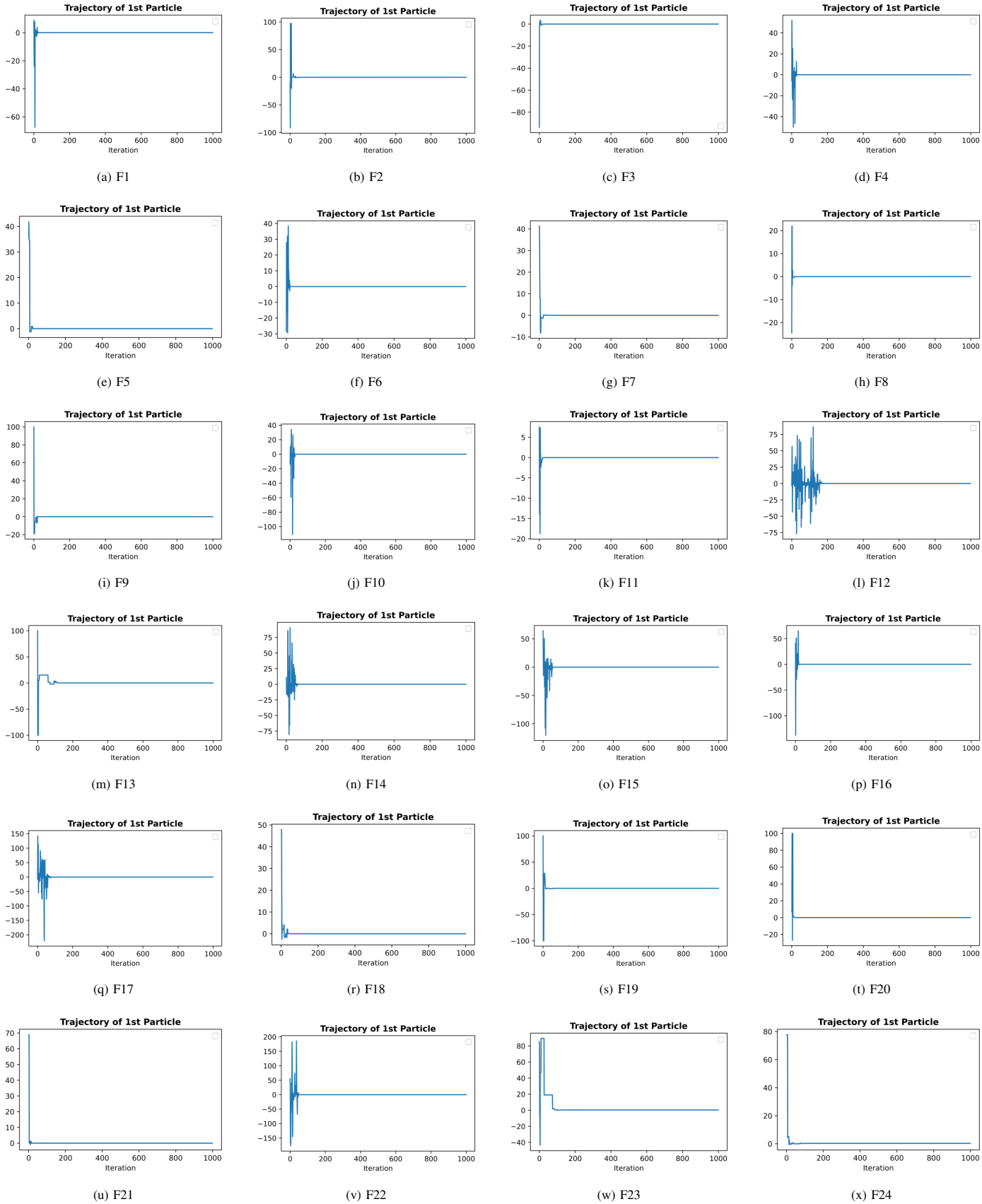
where, u and v are independent random variables following the standard normal distribution; β denotes the Levy flight parameter; L represents the random step size generated from the Levy distribution, exhibiting heavy-tailed characteristics that facilitate large-step.

Integrating the dynamic scaling factors and Levy flight step size, the position updating equation for hen individuals is defined as Eq. (16):

$$\mathbf{x}_{i,j}^{t+1} = \mathbf{x}_{i,j}^t + \lambda_1\, \rho_1\, L\, \left(\mathbf{x}_{l,j}^t - \mathbf{x}_{i,j}^t\right) + \lambda_2\, \rho_2\, L\, \left(\mathbf{x}_{k,j}^t - \mathbf{x}_{i,j}^t\right). \tag{16}$$

where, j denotes the dimension, t the iteration count, and ρ_1 and ρ_1 are random numbers uniformly sampled from the interval [0,1].

By integrating current hen and random individuals' position info with Levy-based step sizes, global exploration and local exploitation are effectively combined. A dynamic scaling factor gives personalized update magnitudes, and the Levy step's long-tail property boosts search jumps, aiding escape from local optima and accelerating global convergence.



(a) F1 (b) F2 (c) F3 (d) F4

(e) F5 (f) F6 (g) F7 (h) F8

(i) F9 (j) F10 (k) F11 (l) F12

(m) F13 (n) F14 (o) F15 (p) F16

(q) F17 (r) F18 (s) F19 (t) F20

(u) F21 (v) F22 (w) F23 (x) F24

Fig. 2. Trajectory of the first particle in ALCSODE during optimization across 24 benchmark functions, illustrating its behavior at different stages.

C. Chick Behavior Updating

In this method, the standard CSO algorithm [33] for updating chick behavior is adopted. For each chick (denoted as individual i), a randomly selected leader individual (indexed as r) serves as the reference during position updating. Randomness and update magnitude are regulated through the following Eq. (17):

$$\mathbf{x}_{i,j}^{t+1} = \mathbf{x}_{i,j}^{t} + \varphi\left(\mathbf{p}_{r,j} - \mathbf{x}_{i,j}^{t}\right). \tag{17}$$

where, $\mathbf{p}_{r,j}$ denotes the historical best position vector of individual r, and φ represents a random scaling factor regulating the update step size, ensuring appropriate stochastic perturbation during updating.

TABLE I. UNIMODAL AND MULTIMODAL TEST FUNCTIONS OF 12 STANDARD BENCHMARKS

No	Functions	Interval	Trait	$f(x)_{min}$
F1	Alpine	[-100,100]	Mulitimodal	0
F2	Elliptic	[-100,100]	Unimodal	0
F3	Schwefel2.22	[-100,100]	Unimodal	0
F4	Sphere	[-100,100]	Unimodal	0
F5	Sum Squares	[-100,100]	Unimodal	0
F6	Ackley	[-100,100]	Mulitimodal	0
F7	Bent Cigar	[-100,100]	Unimodal	0
F8	Discus	[-100,100]	Unimodal	0
F9	Schaffer	[-100,100]	Mulitimodal	0
F10	Quartic	[-100,100]	Unimodal	0
F11	Griewank	[-100,100]	Mulitimodal	0
F12	Rastrigin	[-100,100]	Mulitimodal	0

TABLE II. TEST FUNCTIONS OF CEC2022

No	Functions	Interval	Trait	$f(x)_{min}$
F13	Zakharov	[-100,100]	Unimodal	300
F14	Rosenbrock	[-100,100]	Mulitimodal	400
F15	Schaffer's f6	[-100,100]	Mulitimodal	600
F16	Non-Continuous Rastrigin	[-100,100]	Mulitimodal	800
F17	Levy	[-100,100]	Mulitimodal	900
F18	Hybrid Function 1(N=3)	[-100,100]	Hybrid	1800
F19	Hybrid Function 2(N=6)	[-100,100]	Hybrid	2000
F20	Hybrid Function 3(N=5)	[-100,100]	Hybrid	2200
F21	Composition Function 1(N=5)	[-100,100]	Composition	2300
F22	Composition Function 2(N=4)	[-100,100]	Composition	2400
F23	Composition Function 3(N=5)	[-100,100]	Composition	2600
F24	Composition Function 4(N=6)	[-100,100]	Composition	2700

D. The Elitism Preservation Mechanism

In most CSO and its variant methods, the stochastic nature of individual behavior updating mechanisms (e.g., differentiated strategies for roosters, hens, and chicks) may lead to the loss of high-quality solutions, forcing the algorithm to restart local search and significantly reducing convergence efficiency. To address this, an elitism preservation strategy is introduced to balance exploration-exploitation trade-offs: after each iteration t, individuals with the best fitness are selected to form an elite pool, replacing low-fitness individuals in iteration $t+1$. This mechanism enhances population stability through elite guidance while enabling non-elite individuals to accelerate convergence toward elite solution regions via information sharing, thereby optimizing global convergence performance.

TABLE III. PARAMETER SETTINGS

Algorithm	Parameters
PSO	$c1 = c2 = 2.0$, $w = 0.8$
SHIO	$N_{best} = 3$, $N_{worst} = 1$, $a = 1.5$
INFO	$cond1_t = 0.05$, $cond2_t = 0.05$
CSO	$R_N = 0.2 \times N$, $H_N = 0.6 \times N$, $C_N = N - R_N - H_N$, $M_N = 0.1 \times H_N$, $G = 10$, $F \in$ rand$(0.4, 1)$
L-QCSO	$R_N = 0.2 \times N$, $H_N = 0.6 \times N$, $C_N = N - R_N - H_N$, $M_N = 0.1 \times H_N$, $G = 10$, $F \in$ rand$(0.4, 1)$
ICSO	$R_N = 0.2 \times N$, $H_N = 0.6 \times N$, $C_N = N - R_N - H_N$, $M_N = 0.1 \times H_N$, $G = 10$, $F \in$ rand$(0.4, 1)$, $C = 0.4$
ECSO	$R_N = 0.2 \times N$, $H_N = 0.6 \times N$, $C_N = N - R_N - H_N$, $M_N = 0.1 \times H_N$, $G = 10$, $F \in$ rand$(0.4, 1)$, $k = 1000$
ALCSODE	$R_N = 0.2 \times N$, $H_N = 0.6 \times N$, $C_N = N - R_N - H_N$, $M_N = 0.1 \times H_N$, $G = 10$, $F \in$ rand$(0.4, 1)$, $F_0 = 0.8$,$CR_0 = 0.9$, $\eta = 0.1$

Algorithm 1 Adaptive Levy Flight Chicken Swarm Optimization with Differential Evolution (ALCSODE)

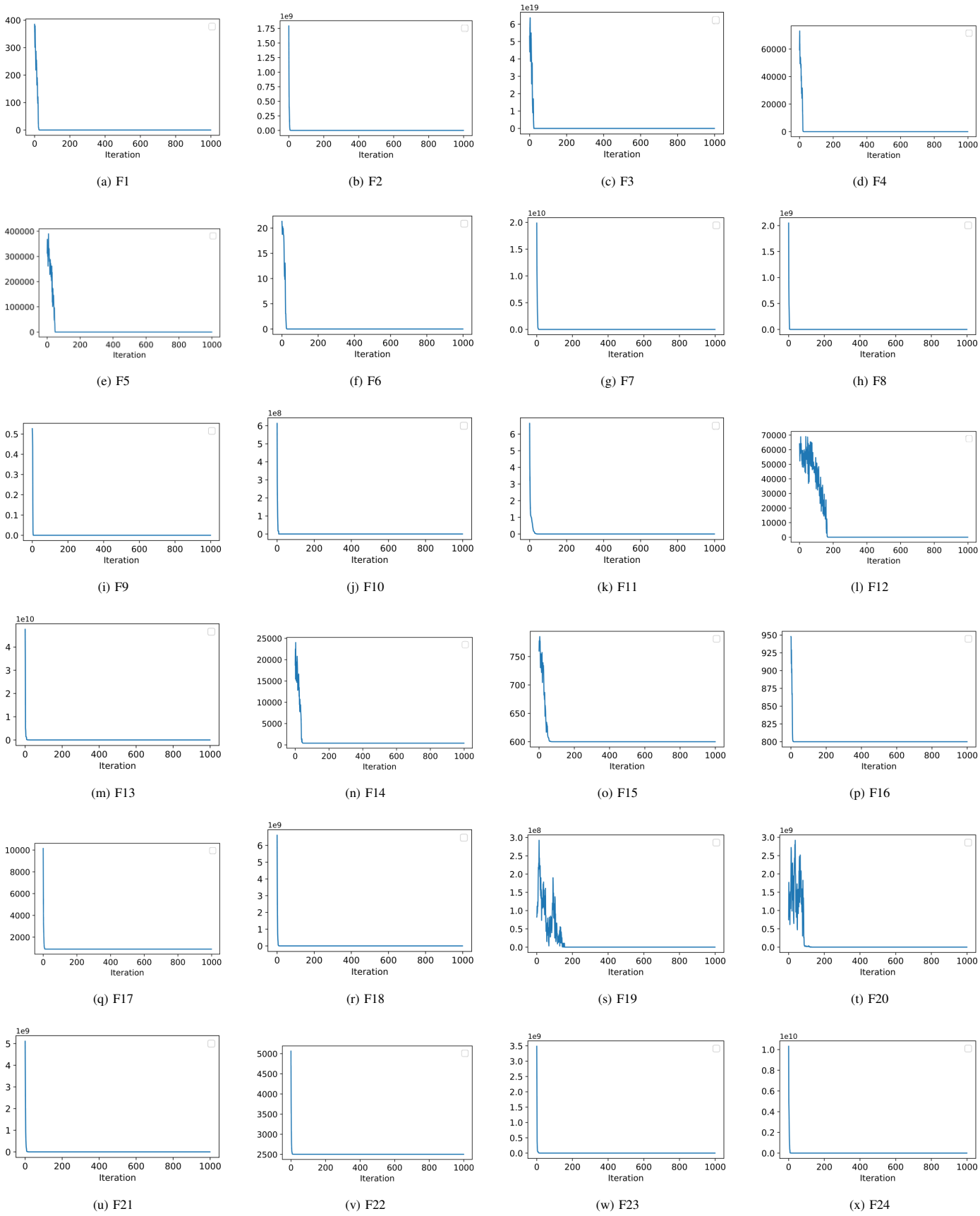
Require: $f(\mathbf{x})$, N, D, T, $F_0, F_{\min}$, $CR_0, CR_{\min}$, θ, η, $[LB, UB]$, β, ε
Ensure: Best solution $\mathbf{x}^*$ and fitness $f(\mathbf{x}^*)$

Init: sample $\{\mathbf{x}_i\}_{i=1}^{N} \sim U[LB, UB]$, eval. f_i, set $\mathbf{x}^*$
for $t = 1 \ldots T$ **do**
  Elite: sort $\{\mathbf{x}_i\}$ by f_i, extract top ηN into E
  Partition remaining into Roosters R, Hens H, Chicks C
  Compute $F_t = F_0(1 - \frac{t}{T})^{\theta} + F_{\min}$, $CR_t = CR_0(1 - \frac{t}{T})^{\theta} + CR_{\min}$
  for all $i \in R$ **do** ▷ Rooster (Adaptive DE + Perturb)
    pick distinct r_1, r_2, r_3, set $\mathbf{d} = \mathbf{x}_{r_1} + F_t(\mathbf{x}_{r_2} - \mathbf{x}_{r_3})$
    $\sigma_i^2 = 1$ if $f_i \le f_{\min}$ else $\exp((f_{\min} - f_i)/(|f_i| + \varepsilon))$
    sample $\delta_i \sim \mathcal{N}(0, \sigma_i^2)$, form $\mathbf{y} = \alpha\mathbf{d} + \delta_i\mathbf{x}_i$
    crossover $[\mathbf{y}, \mathbf{x}_i] \xrightarrow{CR_t} \mathbf{x}_i'$, replace if $f(\mathbf{x}_i') < f_i$
  end for
  for all $i \in H$ **do** ▷ Hen (Levy Flight)
    pick $l, k \ne i$, compute λ_1, λ_2, compute L
    update $\mathbf{x}_i'$, replace if improved
  end for
  for all $i \in C$ **do** ▷ Chick (Follow Leader)
    pick leader r, sample $\varphi \sim U(0, 1)$, set $\mathbf{x}_i' = \mathbf{x}_i + \varphi(\mathbf{p}_r - \mathbf{x}_i)$, replace if $f(\mathbf{x}_i') < f_i$
  end for
  Elite Replacement: inject E into worst slots
  Update global best $\mathbf{x}^*$
end for
  return $\mathbf{x}^*$, $f(\mathbf{x}^*)$

E. Flowchart of the ALCSODE Algorithm

Following the introduction of the four sections above, the detailed pseudocode of the ALCSODE algorithm is presented in Algorithm 1, followed by the overall flowchart illustrated in Fig. 1, the specific steps are then detailed as follows:

Step 1: Problem modeling and algorithm parameters initialization. Define the objective function $f(x)$, population size N, problem dimension j, initial scaling factor F_0, initial crossover



(a) F1 (b) F2 (c) F3 (d) F4

(e) F5 (f) F6 (g) F7 (h) F8

(i) F9 (j) F10 (k) F11 (l) F12

(m) F13 (n) F14 (o) F15 (p) F16

(q) F17 (r) F18 (s) F19 (t) F20

(u) F21 (v) F22 (w) F23 (x) F24

Fig. 3. Average fitness curve of all particles in ALCSODE during optimization, showing the population's overall convergence trend.

TABLE IV. COMPARISON OF EXPERIMENTAL RESULTS FROM F1 TO F8

Function	Algorithm	Mean(dim=10)	Std dev(dim=10)	Best(dim=10)	Mean(dim=50)	Std dev(dim=50)	Best(dim=50)
F1	PSO	1.2171e1	2.8476e1	4.2222	7.4153e2	1.2193e2	7.1206e2
	CSO	8.0693	6.5431	6.2974	1.3867e2	4.0942e1	1.2089e2
	L-QCSO	6.5275	4.3767	6.0746	1.6510e2	2.7145e1	1.5939e2
	ICSO	1.3988	4.4637	5.6363e−1	6.2475e1	5.0660e1	4.1254e1
	ECSO	1.6364e1	5.1912	1.5696e1	2.4910e2	3.6614e1	2.4635e2
	SHIO	3.0936	2.6862	2.2920	1.1520e2	8.4097e1	8.3814e1
	INFO	2.7377	9.2771	3.5371e−4	9.0033	4.0948e1	6.7776e−27
	ALCSODE	**2.4438e-2**	**3.6105e-1**	**5.5180e-174**	**8.6178e-2**	**1.0753**	**4.0827e-165**
F2	PSO	2.0592e7	7.0560e7	1.3823e7	2.3792e9	6.6404e8	2.2575e9
	CSO	6.1484e5	7.0106e6	1.3942e5	6.9277e7	8.0645e7	6.3069e7
	L-QCSO	3.0958e5	4.6929e6	2.5306e3	9.9319e6	6.9314e7	3.4195e6
	ICSO	7.3493e4	1.1988e6	4.5347e3	1.6017e7	6.6362e7	1.0614e6
	ECSO	2.4478e6	2.1051e7	8.5139e5	1.7463e8	1.8639e8	1.6221e8
	SHIO	1.4512e5	3.1824e6	3.0066e−129	8.7830e6	9.7354e7	3.8254e−36
	INFO	2.5434e5	2.4428e6	1.0305e−49	1.0270e7	9.9802e7	7.0123e−48
	ALCSODE	**4.3809e4**	**9.9335e5**	**0.0000**	**1.7922e6**	**4.0895e7**	**0.0000**
F3	PSO	5.0466e10	4.7825e11	4.3917e2	7.8276e77	1.0064e79	2.4860e8
	CSO	2.0637e6	3.2371e7	3.6657e1	1.2770e62	4.0361e63	3.6083e2
	L-QCSO	2.9970e8	6.7707e9	5.1316	4.4075e56	1.3931e58	3.2877e2
	ICSO	2.5172e6	7.9539e7	2.1552e−1	1.6412e62	5.1874e63	5.4136e16
	ECSO	1.0843e6	1.7605e7	7.3739e1	8.0613e60	1.8008e62	1.1096e36
	SHIO	8.1182e8	1.9947e10	2.2337e−82	1.0701e53	1.9500e54	**1.1586e-27**
	INFO	2.7265e7	8.4745e8	8.1567e−27	2.1412e42	6.7677e43	7.6097e−26
	ALCSODE	**1.2155e5**	**3.6764e6**	**3.0275e-246**	**9.8591**	**7.0672e-1**	9.8327
F4	PSO	5.1901e2	3.3705e3	3.6445e1	1.4171e5	1.0273e4	1.3982e5
	CSO	1.4060e2	2.8364e2	1.0768e2	4.0931e3	2.0681e3	3.9001e3
	L-QCSO	2.1161e1	1.3107e2	9.2691	4.4675e3	1.1940e3	4.3806e3
	ICSO	1.4684e1	2.3197e2	4.5753e−3	7.4711e2	3.1091e3	3.0651e−1
	ECSO	2.2854e2	3.7754e2	1.8268e2	9.4319e3	3.4674e3	9.2133e3
	SHIO	7.4621	9.6427e1	5.0667e−133	3.7651e2	3.0527e3	2.3577e−39
	INFO	1.8716e1	2.0239e2	1.3674e−54	1.5604e2	1.5520e3	7.8158e−53
	ALCSODE	**1.5102e-2**	**1.9428e-1**	**5.0988e-321**	**4.8442e-1**	**1.3814e1**	**0.0000**
F5	PSO	2.3160e4	1.3810e4	2.1276e4	2.7388e6	3.2382e5	2.6813e6
	CSO	1.0870e3	1.6029e3	9.5040e2	1.1909e5	5.4754e4	1.1518e5
	L-QCSO	2.8691e2	7.0381e2	2.3292e2	8.2984e4	**3.0516e4**	8.0998e4
	ICSO	1.2231e2	1.6128e3	4.1939e−2	2.1288e4	7.5819e4	1.3021e1
	ECSO	3.1497e3	1.3458e3	3.0072e3	2.7433e5	1.2475e5	2.6596e5
	SHIO	1.2551e1	1.9611e2	9.8009e−130	6.0821e3	4.7958e4	1.5286e−38
	INFO	**9.1802**	**8.0626e1**	7.0286e−54	4.3272e3	3.9028e4	4.4386e−51
	ALCSODE	5.0307e1	1.0947e3	**0.0000**	**1.5263e3**	3.4385e4	**0.0000**
F6	PSO	2.0001e1	**1.7293e-2**	2.0000e1	2.0081e1	3.6345e−2	2.0069e1
	CSO	9.9341	2.8200	8.6036	2.0000e1	8.9650e−1	1.7869e1
	L-QCSO	3.1154	1.8389	2.7988	1.6499e1	7.7940e−1	1.6129e1
	ICSO	7.9773	1.6355	7.7176	2.0072e1	4.3520e−1	1.9810e1
	ECSO	1.4689e1	6.2106e−1	1.4612e1	1.9841e1	**1.8158e-1**	1.9807e1
	SHIO	3.8085	1.7584	3.5751	7.6821	4.9520	5.5864
	INFO	1.0991	3.9546	**4.4409e-16**	6.7263	9.3060	3.9968e−15
	ALCSODE	**3.2380e-2**	3.4513e−1	**4.4409e-16**	**5.2054e-2**	5.5285e−1	**4.4409e-16**
F7	PSO	2.4477e8	1.8254e9	5.0589e4	1.2063e11	1.2824e10	1.1825e11
	CSO	2.4465e8	2.1680e8	2.2719e8	4.6980e9	2.2347e9	4.5431e9
	L-QCSO	4.8091e7	1.8254e8	3.8036e7	3.0407e9	**1.2999e9**	2.9397e9
	ICSO	1.3758e7	1.9220e8	4.9460e3	5.9858e8	3.0465e9	3.1540e5
	ECSO	5.0713e8	2.7607e8	4.8645e8	1.1830e10	3.0300e9	1.1643e10
	SHIO	1.0690e7	1.5594e8	1.9939e−127	2.6346e8	2.1162e9	5.6676e−35
	INFO	1.4142e7	1.7633e8	3.3738e−48	2.1443e8	3.2860e9	1.3765e−46
	ALCSODE	**6.1252e6**	**1.1558e8**	**0.0000**	**6.7484e7**	1.3447e9	**0.0000**
F8	PSO	1.3858e5	1.5930e6	4.9400e4	2.1096e5	1.4863e5	1.9474e5
	CSO	1.3100e4	3.5874e5	5.8645e2	1.1812e4	1.3979e4	1.0421e4
	L-QCSO	7.7057e2	9.0170e3	9.7570e1	3.8709e3	1.6819e4	2.0931e3
	ICSO	4.2348e3	2.7635e3	3.2313e3	2.3510e4	3.5057e4	1.8347e4
	ECSO	1.9217e3	1.7315e3	1.6479e3	1.0926e4	8.4838e3	1.0063e4
	SHIO	**7.5133e1**	**1.0423e3**	2.0267e−126	7.5092e2	**5.7429e3**	3.9283e−39
	INFO	5.9024e3	1.5928e5	7.3494e−50	5.8405e3	2.4436e4	9.0716e−50
	ALCSODE	1.1903e2	2.2259e3	**0.0000**	**4.4433e2**	9.1330e3	**0.0000**

probability CR_0, elite ratio η, maximum iteration times T , and search domain range $[LB, UB]$.

TABLE V. COMPARISON OF EXPERIMENTAL RESULTS FROM F9 TO F16

Function	Algorithm	Mean(dim=10)	Std dev(dim=10)	Best(dim=10)	Mean(dim=50)	Std dev(dim=50)	Best(dim=50)
F9	PSO	2.1915e−3	2.3394e−2	**0.0000**	4.3981e−3	4.0932e−2	**0.0000**
	CSO	1.2950e−3	1.0902e−2	**0.0000**	1.5582e−3	1.5368e−2	1.3605e−11
	L-QCSO	2.8970e−4	4.1479e−3	**0.0000**	7.0125e−4	1.1484e−2	**0.0000**
	ICSO	1.1525e−3	1.3775e−2	9.9178e−10	5.6907e−4	1.0126e−2	2.4241e−11
	ECSO	2.0857e−3	1.8151e−2	2.5352e−6	1.5202e−3	8.8111e−3	7.5935e−8
	SHIO	6.7496e−4	2.6408e−3	**0.0000**	9.3982e−4	7.2033e−3	**0.0000**
	INFO	2.0511e−4	3.5885e−3	**0.0000**	1.2323e−3	1.1032e−2	**0.0000**
	ALCSODE	**3.8614e-5**	**1.2176e-3**	**0.0000**	**2.8502e-7**	**8.4581e-6**	**0.0000**
F10	PSO	9.8715e6	8.7741e7	1.0265e4	1.5410e10	2.7870e9	1.4932e10
	CSO	8.5630e4	1.5200e6	8.9011e2	1.4151e7	1.1050e8	8.4274e6
	L-QCSO	2.3330e4	**3.7674e5**	1.9332e−1	9.5990e6	6.3321e7	5.5855e6
	ICSO	6.1236e4	1.8500e6	1.5713e−3	2.0692e7	2.2624e8	2.4192e−1
	ECSO	2.6265e5	1.6985e6	1.5588e5	4.9450e7	2.0985e8	3.4797e7
	SHIO	5.3305e4	9.7508e5	**2.1546e-5**	1.1907e7	1.3905e8	2.6926e−3
	INFO	2.5538e4	4.3033e5	3.8682e−4	2.4971e6	3.4058e7	**1.1924e-3**
	ALCSODE	**2.1052e4**	5.5612e5	1.0986e−4	**1.3696e2**	**3.1244e2**	1.2381e2
F11	PSO	2.2655	6.9911e−1	2.1518	2.7139e1	3.5048	2.6552e1
	CSO	4.0950e−1	1.2417e−1	3.9084e−1	2.0201	6.0190e−1	1.9773
	L-QCSO	6.4440e−1	6.0465e−2	6.3673e−1	2.2367	2.4165e−1	2.2226
	ICSO	1.0660e−1	1.5407e−1	8.4411e−2	4.2695e−1	1.0284	2.2509e−2
	ECSO	1.1209	8.1574e−2	1.1151	2.3045	5.6463e−1	2.2648
	SHIO	1.6011e−1	1.2006e−1	1.1470e−1	2.0404e−1	1.0539	**0.0000**
	INFO	1.5093e−1	2.8325e−1	2.4612e−2	2.6216e−1	5.6060e−1	**0.0000**
	ALCSODE	**1.3474e-3**	**2.7760e-2**	**0.0000**	**1.5361e-3**	**2.9620e-2**	**0.0000**
F12	PSO	4.0863e2	3.3017e3	2.3099e1	1.3827e5	1.2042e4	1.3592e5
	CSO	4.8146e2	2.9826e2	4.5710e2	3.7476e3	2.0547e3	3.5948e3
	L-QCSO	1.4756e2	1.5853e2	1.2427e2	7.2951e3	1.0877e3	7.2002e3
	ICSO	3.0308e2	1.7108e2	2.8938e2	4.8178e3	2.5093e3	4.4025e3
	ECSO	5.0184e2	5.2995e2	4.6734e2	7.1712e3	3.7697e3	6.9471e3
	SHIO	5.3312e1	1.9080e2	3.5870e1	1.1539e3	2.1315e3	8.4338e2
	INFO	2.7375e1	2.0187e2	**0.0000**	2.2399e2	1.1400e3	**0.0000**
	ALCSODE	**6.2019**	**1.2584e2**	**0.0000**	**1.3460e2**	**2.4440e1**	1.3308e2
F13	PSO	1.4231e4	4.4013e3	1.3391e4	1.5764e5	2.2018e4	1.5286e5
	CSO	8.5161e2	1.0744e3	7.7142e2	2.6140e7	7.9264e8	7.5380e3
	L-QCSO	2.9007e3	2.8382e3	2.6788e3	1.2193e4	1.6243e4	1.0348e4
	ICSO	5.9110e2	1.3715e3	3.0002e2	3.9556e6	1.1438e8	1.4113e4
	ECSO	1.1083e3	3.7852e2	1.0808e3	1.3757e4	8.8871e4	9.2386e3
	SHIO	3.5886e2	3.8100e2	**3.0000e2**	1.9168e4	1.0995e5	6.8916e2
	INFO	6.3154e2	2.5764e3	**3.0000e2**	2.4158e3	6.6430e3	**3.0000e2**
	ALCSODE	**3.5761e2**	**1.2225e3**	3.0201e2	**4.1672e2**	**3.1668e3**	3.0686e2
F14	PSO	4.3204e2	2.2300e2	4.0918e2	2.2040e4	4.6964e3	2.1181e4
	CSO	4.2989e2	2.3438e1	4.2754e2	1.3786e3	4.1279e2	1.3488e3
	L-QCSO	4.0247e2	2.6004e1	4.0011e2	6.9569e2	2.9382e2	6.3642e2
	ICSO	4.0208e2	2.2963e1	4.0030e2	5.5840e2	4.9277e2	4.1015e2
	ECSO	4.4100e2	4.1539e1	4.1380e2	1.6100e3	6.0620e2	1.5690e3
	SHIO	4.0976e2	1.1963e1	4.0665e2	4.8056e2	6.2838e2	**4.0000e2**
	INFO	4.0083e2	**5.9497**	**4.0000e2**	4.3623e2	3.8766e2	**4.0000e2**
	ALCSODE	**4.0065e2**	1.0023e1	**4.0000e2**	**4.1052e2**	**1.6400e2**	**4.0000e2**
F15	PSO	6.3579e2	8.2919	6.3446e2	6.8799e2	1.0618e1	6.8505e2
	CSO	6.0600e2	3.2536	6.0528e2	6.1918e2	3.2556	6.1854e2
	L-QCSO	6.0093e2	2.3546	6.0060e2	6.2087e2	**2.2176**	6.2060e2
	ICSO	6.0352e2	2.3401	6.0324e2	6.1784e2	4.0763	6.1697e2
	ECSO	6.1522e2	2.2993	6.1402e2	6.2467e2	3.4395	6.2439e2
	SHIO	6.0255e2	2.7280	6.0208e2	6.1064e2	6.0517	6.0847e2
	INFO	6.0070e2	2.5381	**6.0000e2**	6.0121e2	5.1117	**6.0000e2**
	ALCSODE	**6.0015e2**	**2.0986**	**6.0000e2**	**6.0016e2**	2.3491	**6.0000e2**
F16	PSO	8.2662e2	5.4037	8.2600e2	1.1815e3	2.3858e1	1.1780e3
	CSO	8.0007e2	7.6322e−1	**8.0000e2**	8.1058e2	6.1036	8.1000e2
	L-QCSO	8.0005e2	5.4844e−1	**8.0000e2**	8.1039e2	**3.2107**	8.1000e2
	ICSO	8.0004e2	6.4170e−1	**8.0000e2**	8.1348e2	6.6564	8.1300e2
	ECSO	8.0008e2	1.1140	**8.0000e2**	8.2844e2	6.9513	8.2800e2
	SHIO	8.0001e2	**1.4784e-1**	**8.0000e2**	8.0102e2	6.2935	**8.0000e2**
	INFO	8.2100e2	8.6180e−1	8.1200e2	8.1040e2	4.2054	8.1000e2
	ALCSODE	**8.0001e2**	3.0299e−1	**8.0000e2**	**8.0021e2**	4.3630	**8.0000e2**

TABLE VI. COMPARISON OF EXPERIMENTAL RESULTS FROM F17 TO F24

Function	Algorithm	Mean(dim=10)	Std dev(dim=10)	Best(dim=10)	Mean(dim=50)	Std dev(dim=50)	Best(dim=50)
F17	PSO	2.6326e3	5.6572e2	2.5372e3	2.7243e4	8.5072e3	2.4246e4
	CSO	9.2941e2	8.7053e1	9.2097e2	1.8570e3	7.6056e2	1.7195e3
	L-QCSO	9.0734e2	5.1628e1	9.0183e2	1.8656e3	**3.1121e2**	1.8282e3
	ICSO	9.2844e2	6.9121e1	9.2023e2	1.5213e3	9.5674e2	1.4210e3
	ECSO	9.5393e2	1.3974e2	9.4287e2	3.0568e3	9.3457e2	2.9835e3
	SHIO	9.0695e2	5.4043e1	9.0026e2	4.5717e3	1.2557e3	3.8546e3
	INFO	9.0316e2	4.5299e1	**9.0000e2**	9.8810e2	7.7951e2	9.0032e2
	ALCSODE	**9.0121e2**	**2.6958e1**	**9.0000e2**	**9.2830e2**	5.4272e2	**9.0000e2**
F18	PSO	1.5642e7	1.7129e8	1.0142e4	2.1166e10	5.6262e9	2.0352e10
	CSO	3.8821e5	4.3947e6	1.8530e3	1.4970e9	7.8855e8	1.4391e9
	L-QCSO	7.7114e5	1.4315e7	1.8059e3	2.6978e9	5.0931e8	2.6678e9
	ICSO	**1.5312e5**	**2.7530e6**	2.1249e3	4.6877e7	7.9032e8	4.0565e4
	ECSO	2.0135e6	2.2907e7	2.1130e3	1.1210e9	1.5212e9	1.0253e9
	SHIO	4.8034e5	1.4679e7	**1.8004e3**	7.0391e7	7.6071e8	1.8015e3
	INFO	9.7803e5	1.2583e7	1.8005e3	3.8950e7	3.7818e8	1.8015e3
	ALCSODE	3.1144e5	9.1831e6	**1.8004e3**	**8.8508e6**	**1.7131e8**	**1.8014e3**
F19	PSO	2.0669e3	2.1771e1	2.0637e3	1.1338e4	7.4889e3	1.0181e4
	CSO	2.0217e3	1.7799e1	2.0194e3	2.9263e3	2.0822e3	2.7509e3
	L-QCSO	2.0083e3	5.8092e1	2.0038e3	3.3907e3	4.2409e3	3.0374e3
	ICSO	2.0476e3	2.5859e1	2.0457e3	2.5364e3	**1.1565e3**	2.1599e3
	ECSO	2.0585e3	6.9333e1	2.0415e3	3.1794e3	4.5674e3	2.9015e3
	SHIO	2.0213e3	**1.6653e1**	2.0190e3	3.4178e3	4.1336e3	2.1328e3
	INFO	2.0040e3	2.8061e1	**2.0000e3**	2.5296e3	3.1631e3	**2.0009e3**
	ALCSODE	**2.0026e3**	6.0033e1	2.0006e3	**2.1070e3**	3.0972e3	**2.0009e3**
F20	PSO	3.1400e3	2.2436e3	2.9085e3	3.9415e4	1.7860e4	3.6786e4
	CSO	2.2278e3	7.7401e1	2.2209e3	5.9860e3	7.0967e4	3.5996e3
	L-QCSO	2.2176e3	1.5813e2	2.2093e3	4.7364e3	9.5444e3	3.9668e3
	ICSO	2.2103e3	1.1385e2	2.2038e3	4.5770e3	2.9931e3	2.6551e3
	ECSO	2.3216e3	1.6637e2	2.3078e3	4.7201e3	2.9933e3	4.4365e3
	SHIO	2.2094e3	**4.3933e1**	2.2015e3	2.9708e3	**1.5570e3**	2.3036e3
	INFO	2.2117e3	8.6726e1	2.2001e3	4.4880e3	1.0877e4	2.2021e3
	ALCSODE	**2204.03632**	7.9725e1	**2.2000e3**	**2.7890e3**	2.4339e3	**2.2004e3**
F21	PSO	1.2330e8	7.9524e8	6.5777e6	3.1866e10	2.3750e9	3.1437e10
	CSO	8.9897e7	7.8170e7	8.3546e7	1.1761e9	5.2367e8	1.1365e9
	L-QCSO	1.2882e7	5.8898e7	1.5843e6	7.8225e8	2.6790e8	7.6479e8
	ICSO	2.7389e6	5.7325e7	4.2223e3	1.7390e8	7.6455e8	7.0218e4
	ECSO	3.2675e7	9.7277e7	2.5080e7	1.4227e9	6.9794e8	1.3683e9
	SHIO	3.7806e6	5.4935e7	**2.5000e3**	9.2896e7	7.0122e8	**2.5000e3**
	INFO	3.3374e6	3.4619e7	**2.5000e3**	4.0015e7	4.0280e8	**2.5000e3**
	ALCSODE	**5.4514e5**	**1.0724e7**	**2.5000e3**	**1.4465e7**	**2.7567e8**	**2.5000e3**
F22	PSO	2.6580e3	2.3430e2	2.6236e3	6.7897e3	2.7290e3	6.0614e3
	CSO	2.5191e3	2.4691e1	2.5159e3	2.9446e3	3.1401e2	2.8995e3
	L-QCSO	2.5091e3	1.6583e1	2.5061e3	2.9955e3	**1.2763e2**	2.9810e3
	ICSO	2.5139e3	2.2874e1	2.5113e3	2.7975e3	3.0825e2	2.7530e3
	ECSO	2.5249e3	4.2371e1	2.5226e3	3.2843e3	4.9190e2	3.2497e3
	SHIO	2.5127e3	**1.5887e1**	2.5104e3	3.5894e3	4.6065e2	3.4663e3
	INFO	2.5082e3	2.3643e1	**2.5030e3**	2.5465e3	2.0419e2	**2.5156e3**
	ALCSODE	**2.5039e3**	1.8663e1	**2.5030e3**	**2.5247e3**	1.6702e2	**2.5156e3**
F23	PSO	7.0138e3	4.7479e4	5.0129e3	9.9541e4	9.9534e5	3.8218e4
	CSO	3.1115e3	5.3457e2	3.0597e3	1.7654e4	2.5887e5	5.8710e3
	L-QCSO	3.6831e3	6.8820e2	3.1785e3	4.5210e3	5.2698e3	3.8771e3
	ICSO	2.9424e3	**3.0850e2**	**2.8207e3**	9.3559e4	2.5690e6	9.3078e3
	ECSO	3.1238e3	3.2901e2	3.0977e3	5.8154e3	**1.8505e3**	5.6269e3
	SHIO	2.9200e3	4.3141e2	2.8208e3	6.1869e3	5.3195e3	2.8485e3
	INFO	2.9486e3	8.6845e2	**2.8207e3**	1.0965e4	5.4872e4	2.8236e3
	ALCSODE	**2.9199e3**	3.0715e3	**2.8207e3**	**3.1587e3**	7.3032e3	**2.8218e3**
F24	PSO	7.8219e7	5.9782e8	4.7312e4	9.0122e9	3.6756e9	8.3934e9
	CSO	3.0742e7	5.4115e7	2.6376e7	7.6600e8	4.7189e8	7.2143e8
	L-QCSO	2.0039e7	3.6293e7	5.3834e6	1.1190e9	4.8782e8	1.0777e9
	ICSO	3.0929e6	4.9371e7	1.2905e4	3.5972e8	9.6735e8	5.8719e4
	ECSO	4.8070e7	5.1804e7	4.0272e7	1.0143e9	1.0010e9	9.3863e8
	SHIO	1.7249e6	2.2613e7	**2.9522e3**	1.4510e8	8.2402e8	**2.9614e3**
	INFO	2.2004e6	**1.8753e7**	**2.9522e3**	**4.0366e7**	**2.9775e8**	**2.9614e3**
	ALCSODE	**9.6401e5**	2.5747e7	**2.9522e3**	5.8906e7	5.0165e8	**2.9614e3**

TABLE VII. WILCOXON TEST RESULTS FOR BEST VALUES WITH FIXED ITERATIONS

	Dim	PSO		CSO		L-QCSO		ICSO		ECSO		SHIO		INFO		ALCSODE
		Wilcox	Rank	Wilcox	Rank	Wilcox	Rank	Wilcox	Rank	Wilcox	Rank	Wilcox	Rank	Wilcox	Rank	Rank
F1	10	+	5	+	7	+	6	+	3	+	8	+	4	+	2	1
	50	+	8	+	5	+	6	+	3	+	7	+	4	+	2	1
F2	10	+	8	+	6	+	4	+	5	+	7	+	2	+	3	1
	50	+	8	+	6	+	5	+	4	+	7	+	3	+	2	1
F3	10	+	8	+	6	+	5	+	4	+	7	+	2	+	3	1
	50	+	7	+	5	-	4	+	6	+	8	-	1	-	2	3
F4	10	+	6	+	7	+	5	+	4	+	8	+	2	+	3	1
	50	+	8	+	5	+	6	+	4	+	7	+	3	+	2	1
F5	10	+	8	+	6	+	5	+	4	+	7	+	2	+	3	1
	50	+	8	+	6	+	5	+	4	+	7	+	3	+	2	1
F6	10	+	8	+	6	+	3	+	5	+	7	+	4	=	1	1
	50	+	8	+	5	+	4	+	7	+	6	+	3	+	2	1
F7	10	+	5	+	7	+	6	+	4	+	8	+	2	+	3	1
	50	+	8	+	6	+	5	+	4	+	7	+	3	+	2	1
F8	10	+	8	+	5	+	4	+	7	+	6	+	2	+	3	1
	50	+	8	+	6	+	4	+	7	+	5	+	3	+	2	1
F9	10	=	1	=	1	=	1	+	7	+	8	=	1	=	1	1
	50	=	1	+	6	=	1	+	7	+	8	=	1	=	1	1
F10	10	+	7	+	6	+	5	+	4	+	8	-	1	+	3	2
	50	+	8	+	6	+	5	-	3	+	7	-	2	-	1	4
F11	10	+	8	+	5	+	6	+	3	+	7	+	4	+	2	1
	50	+	8	+	5	+	6	+	4	+	7	=	1	=	1	1
F12	10	+	3	+	7	+	5	+	6	+	8	+	4	=	1	1
	50	+	8	+	4	+	7	+	5	+	6	+	3	-	1	2
F13	10	+	8	+	5	+	7	+	4	+	6	-	1	-	1	3
	50	+	8	+	4	+	6	+	7	+	5	+	3	-	1	2
F14	10	+	6	+	8	+	3	+	4	+	7	+	5	=	1	1
	50	+	8	+	6	+	5	+	4	+	7	=	1	=	1	1
F15	10	+	8	+	6	+	3	+	5	+	7	+	4	=	1	1
	50	+	8	+	5	+	6	+	4	+	7	+	3	=	1	1
F16	10	+	8	=	1	=	1	=	1	=	1	=	1	+	7	1
	50	+	8	+	2	+	2	+	6	+	7	=	1	+	2	1
F17	10	+	8	+	6	+	4	+	5	+	7	+	3	=	1	1
	50	+	8	+	4	+	5	+	3	+	6	+	7	+	2	1
F18	10	+	8	+	5	+	4	+	7	+	6	=	1	+	3	1
	50	+	8	+	6	+	7	+	4	+	5	+	2	+	2	1
F19	10	+	8	+	4	+	3	+	7	+	6	+	4	-	1	2
	50	+	8	+	5	+	7	+	4	+	6	+	3	=	1	1
F20	10	+	8	+	6	+	5	+	4	+	7	+	3	+	2	1
	50	+	8	+	6	+	5	+	4	+	7	+	3	+	2	1
F21	10	+	6	+	8	+	5	+	4	+	7	=	1	=	1	1
	50	+	8	+	7	+	5	+	4	+	6	=	1	=	1	1
F22	10	+	8	+	6	+	3	+	5	+	7	+	4	=	1	1
	50	+	8	+	4	+	5	+	3	+	6	+	7	=	1	1
F23	10	+	8	+	5	+	7	=	1	+	6	+	4	=	1	1
	50	+	8	+	6	+	4	+	7	+	5	+	3	+	2	1
F24	10	+	5	+	7	+	6	+	4	+	8	=	1	=	1	1
	50	+	8	+	5	+	7	+	4	+	6	=	1	=	1	1
Avg			7.25		5.43		4.75		4.56		6.64		2.58		1.77	1.22
Final			8		6		5		4		7		3		2	1

Step 2: Population initialization based on the normal distribution, calculate the fitness using the objective function $f(x)$, and record the optimal value.

Step 3: Elite selection. Select top $\eta * N$ individuals. The residual population is categorized into roosters, hens, and chicks based on fitness hierarchy.

Step 4: Adaptive perturbation-based differential evolution for rooster behavior updating. First, the adaptive differential mutation vector is computed using the adaptive scaling factor F_t through Eq. (7). Then, the individual-adaptive perturbation factor δ_i is calculated, where inferior individuals receive larger perturbations through a fitness-dependent variance σ_i^2, sampled from a zero-mean normal distribution as defined in Eq. (9). Finally, the differential evolution mutation vector is combined with the current rooster vector controlled by the adaptive perturbation factor δ_i to generate the next generation, as defined in Eq. (10). Additionally, a component-wise crossover strategy prevents excessive localization by updating positions dimension-by-dimension, as defined in Eq. (11).

Step 5: Levy flight-based hen behavior optimization. By leveraging the Levy flight algorithm concept, hen behavior is updated in Equation 16 through the integration of dynamic scaling factors λ_1, λ_2 and Levy flight step size L.

Step 6: Chick behavior updating. For each chick i, a randomly selected leader r serves as the reference for position updating in Eq. (17).

Step 7: The fitness values of the population in the new state are evaluated by $f(x)$, with updates occurring only if the new fitness is better than the previous value. Additionally, the elite strategy replaces the current worst-performing individuals.

Step 8: Termination condition. The algorithm terminates if the maximum iteration count is reached, outputting the optimal solution; otherwise, the process returns to Step 3.

F. Time Complexity Analysis

The time complexity of the proposed Adaptive Levy Flight Chicken Swarm Optimization with Differential Evolution algorithm is analyzed with respect to the population size N, problem dimensionality Dim, fitness evaluation cost $f(Dim)$, and the maximum number of iterations T. In each iteration, the algorithm updates the positions of all individuals—including roosters, hens, and chicks—through vector operations such as mutation, adaptive perturbation, crossover, and Levy flight-based steps, each requiring $O(Dim)$ computations per individual and resulting in an overall $O(N \times Dim)$ complexity. Fitness evaluations for all individuals contribute an additional cost of $O(N \times f(Dim))$ per iteration. Although elitism preservation involves selecting elite individuals and replacing low-fitness ones, which may require sorting with a complexity

of $O(N \log N)$, this cost is generally minor compared to the dominant vector updates and fitness evaluations. Consequently, the total time complexity per iteration can be approximated by combining these main components as $O(N \times Dim) + O(N \times f(Dim))$, and over T iterations, the overall time complexity is $O\big(T \times N \times (Dim + f(Dim))\big)$. This formulation highlights that the computational cost of ALCSODE scales linearly with population size, problem dimensionality, and the complexity of fitness evaluation, demonstrating its suitability for large-scale and high-dimensional optimization problems.

IV. Simulation Experiment

To validate the effectiveness of the proposed ALSCODE algorithm, extensive experiments are conducted on 24 benchmark functions, including unimodal functions, high-dimensional multimodal functions, and the CEC2022 test suite [47]. This section first presents the experimental configurations and parameter settings of the comparative methods, followed by an analysis and verification of the algorithm's effectiveness based on the experimental results.

A. Experimental Setup

The experiments are conducted on a Windows 11 system equipped with a 2.9 GHz CPU and 16 GB of RAM, using Python 3.10 as the programming environment. The ALSCODE algorithm is evaluated on 24 benchmark functions, including 8 unimodal functions, 9 multimodal functions, 3 hybrid functions, and 4 composition functions. Detailed information, including function names, search ranges, and theoretical global optima, is provided in Table I and Table II, the latter corresponding to the standard CEC2022 benchmark.

The experimental comparison includes the standard Particle Swarm Optimization (PSO) algorithm inspired by bird flocking behavior [48], the standard Chicken Swarm Optimization (CSO) algorithm [33], along with three of its variants (L-QCSO [39], ICSO [36], and ECSO [49]), as well as the metaheuristic algorithms INFO [12] and SHIO [13], which serve as baseline methods. To ensure a fair comparison, all algorithms adopt the same configuration: a population size of 50, a maximum of 1000 iterations, and search spaces of 10 and 50 dimensions. Each algorithm is executed 30 times on each benchmark function, and the average performance is recorded. Additional parameter settings are provided in Table III.

B. Qualitative Results and Discussion of ALCSODE

The qualitative experiments are conducted on the two-dimensional versions of benchmark functions to visually observe the search behavior of the ALSCODE algorithm. For each test function, two types of diagrams are provided: one illustrating the trajectory of the first particle in the first dimension, and the other depicting the average fitness of the particle swarm. The former reflects the search dynamics of a representative particle, while the latter indicates the convergence trend of the entire population.

Fig. 2 shows the trajectory of the first particle in the first dimension, which serves to evaluate the algorithm's behavior at different stages, particularly whether it exhibits abrupt movements in the early iterations and stabilizes in later stages. This figure consists of 24 subplots, each representing the trajectory corresponding to a specific benchmark function. For example, Fig. 2a illustrates the trajectory for the F1 function (Alpine). According to the study by Berg et al. [50], an effective population-based algorithm performs wide-range exploration in the early phase and gradually shifts to local exploitation to achieve high-quality solutions. As shown in the figure, the first particle in ALSCODE exhibits a significant and abrupt movement exceeding 50% of the search space in the early phase, indicating that the algorithm fully utilizes population diversity and individual differences to escape local optima and conduct global exploration. As the iterations proceed, the particle's movement amplitude gradually decreases and the trajectory becomes smoother. This progressive stabilization is attributed to ALSCODE's adaptive perturbation mechanism, which dynamically adjusts the step size of the movement vector based on particle fitness, thereby narrowing the search range and guiding particles toward more promising regions for enhanced local exploitation.

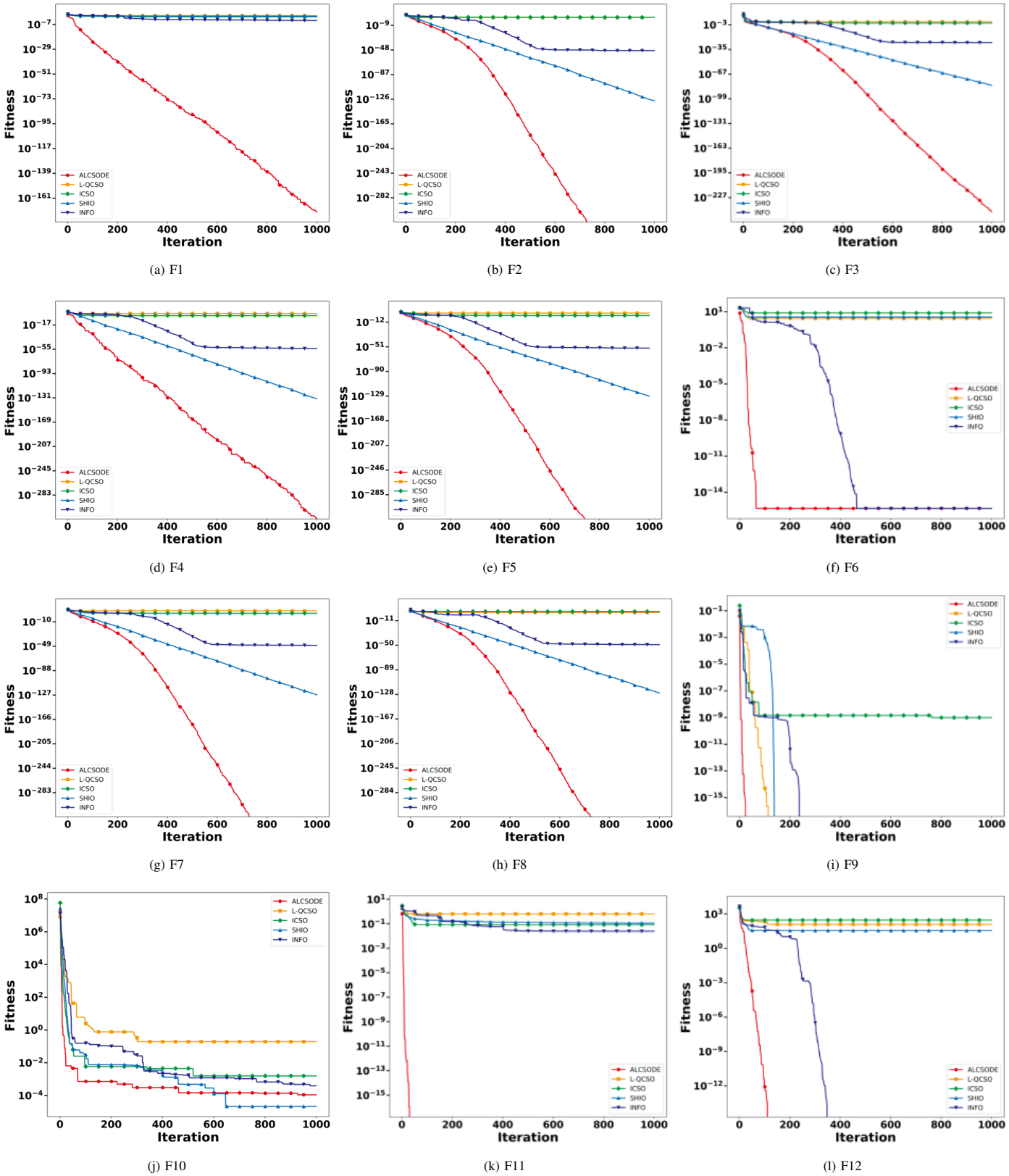
The convergence curves of average fitness in Fig. 3 further validate this behavior. All test functions exhibit a consistently decreasing trend, indicating that the overall solution quality of the population improves continuously throughout the optimization process. This result confirms that ALSCODE effectively optimizes the swarm's search behavior and significantly enhances the accuracy of the approximated optimal solutions.

In summary, Fig. 2 and Fig. 3 jointly demonstrate that ALSCODE maintains a well-balanced dynamic between exploration and exploitation. The large-scale fluctuations in the early stage enhance diversity and global coverage, while the gradually decreasing fluctuations in the later stage ensure convergence and fine-tuning. Ultimately, the particle trajectories on nearly all benchmark functions converge to a stable point, verifying that ALSCODE achieves a smooth transition from global exploration to local exploitation and exhibits strong convergence performance and robustness.

C. ALCSODE Statistical Results and Comparison with Other Algorithms

Functions F1 to F24 (including the CEC2022 test suite) are evaluated under 10-dimensional and 50-dimensional settings. Experiments are based on data obtained from 30 independent runs for each function, and performance metrics include the best value, mean value, and standard deviation. The results are summarized in Table IV, Table V, and Table VI, where the best results for each function are highlighted in bold.

To comprehensively validate the performance of the ALSCODE algorithm, it is compared with the classical Chicken Swarm Optimization (CSO) and its three variants L-QCSO, ICSO, and ECSO, as well as representative metaheuristic algorithms from outside the domain, including PSO, INFO, and SHIO. Fig. 4 to Fig. 7 present the convergence curves of a randomly selected run among the 30 executions. To reduce visual clutter, only the top-performing or representative algorithms are shown: L-QCSO and ICSO among chicken swarm algorithms, and INFO and SHIO among metaheuristic algorithms. This choice facilitates a clearer comparison of convergence behavior. Comparative results indicate that, under different dimensional conditions, ALSCODE exhibits significant advantages in solution accuracy, result stability, and convergence efficiency.



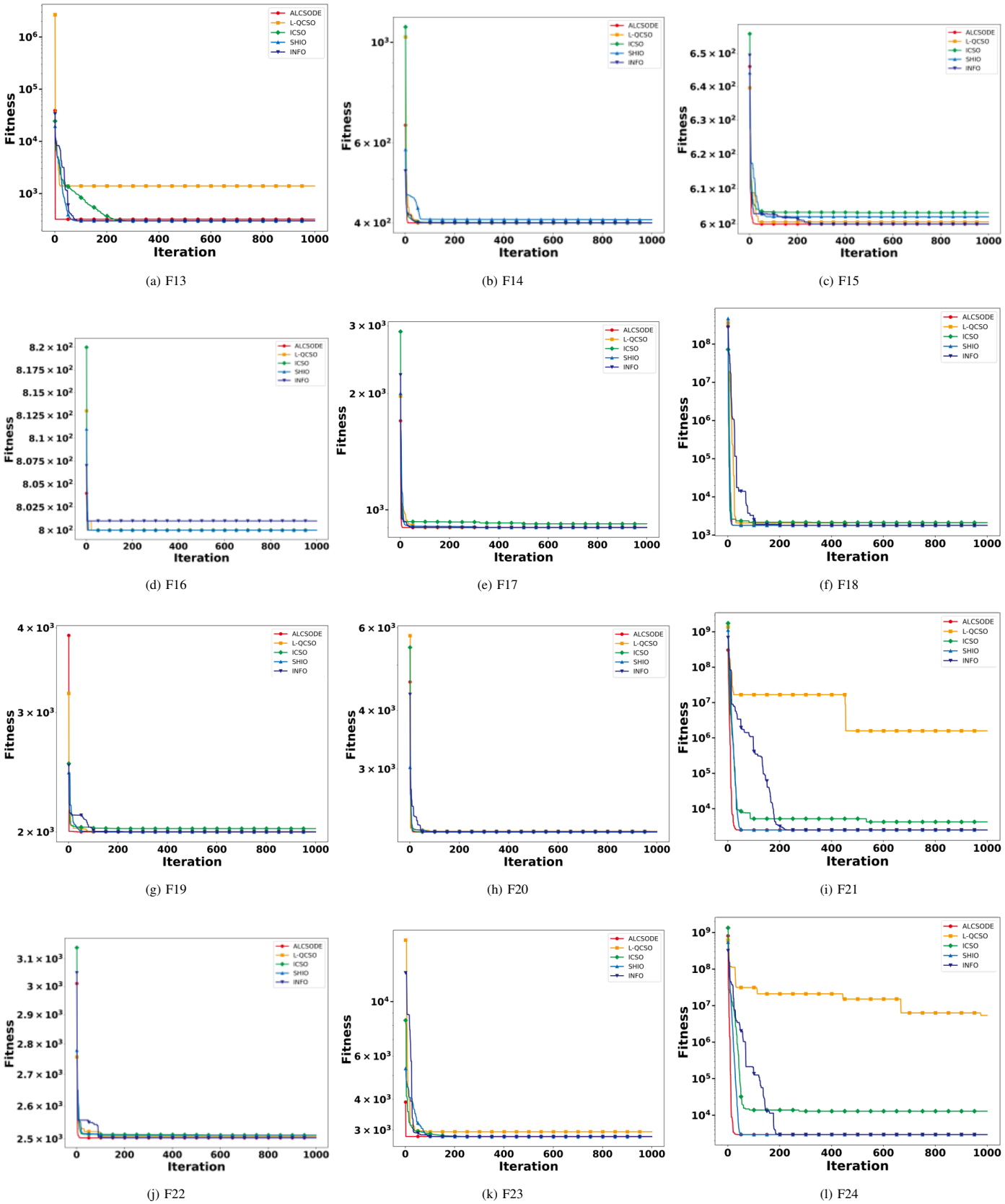
(a) F1 (b) F2 (c) F3

(d) F4 (e) F5 (f) F6

(g) F7 (h) F8 (i) F9

(j) F10 (k) F11 (l) F12

Fig. 4. Performance comparison of convergence curve on F1-F12 test functions in 10D.



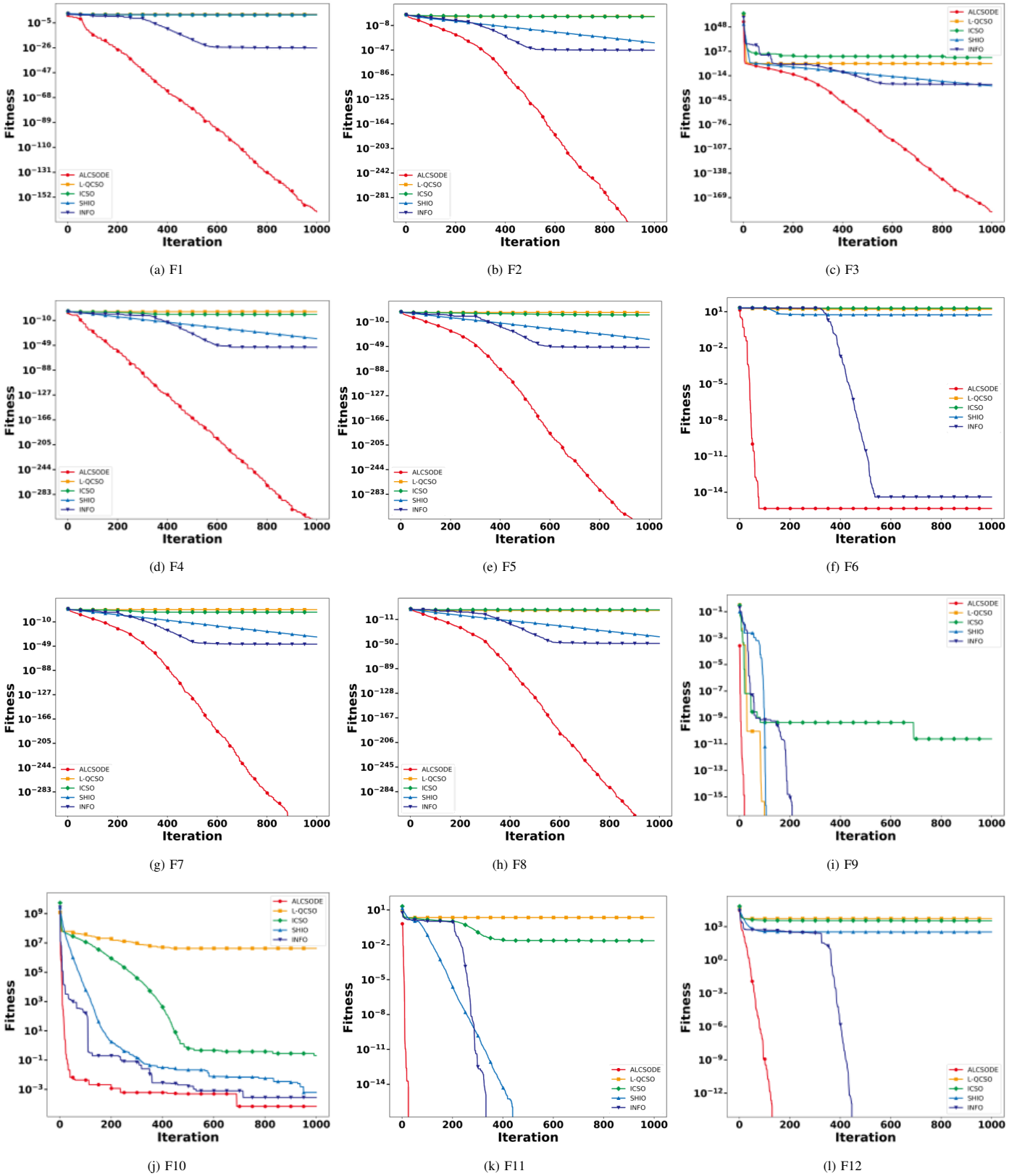
(a) F13 (b) F14 (c) F15

(d) F16 (e) F17 (f) F18

(g) F19 (h) F20 (i) F21

(j) F22 (k) F23 (l) F24

Fig. 5. Performance comparison of convergence curve on F13-F24 test functions in 10D.



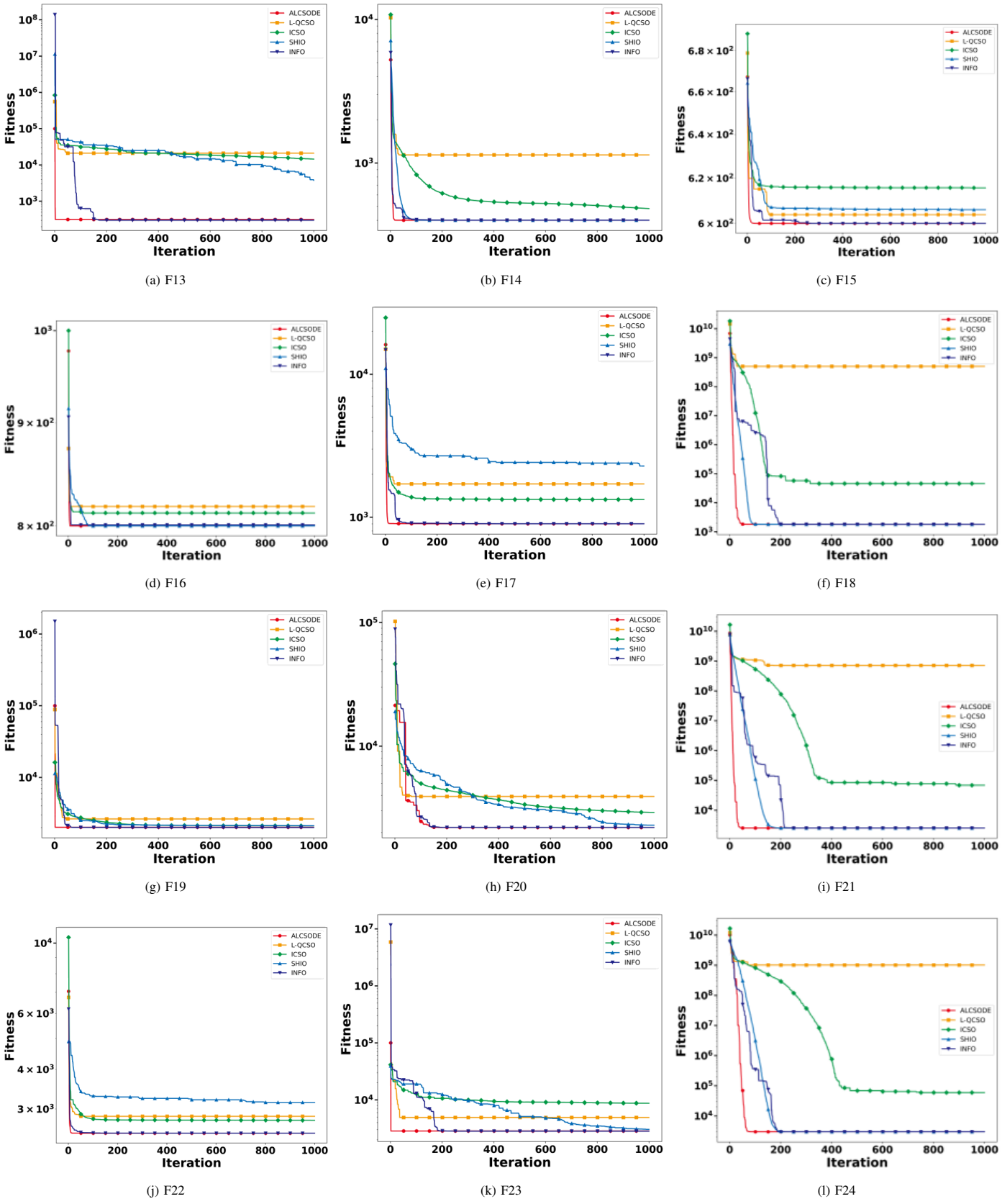
(a) F1 (b) F2 (c) F3

(d) F4 (e) F5 (f) F6

(g) F7 (h) F8 (i) F9

(j) F10 (k) F11 (l) F12

Fig. 6. Performance comparison of convergence curve on F1-F12 test functions in 50D.



(a) F13 (b) F14 (c) F15

(d) F16 (e) F17 (f) F18

(g) F19 (h) F20 (i) F21

(j) F22 (k) F23 (l) F24

Fig. 7. Performance comparison of convergence curve on F13-F24 test functions in 50D.

1) Evaluation of unimodal banchmark: Unimodal functions F2, F3, F4, F5, F7, F8, F10, and F13 contain only a single global optimum and no local optima, making them ideal benchmarks for evaluating the development capabilities of optimization algorithms. According to the comparison of Mean, Standard Deviation, and Best results in both 10-dimensional and 50-dimensional cases, ALCSODE successfully reaches the global optimum on F2, F5, and F8, and demonstrates the best performance on all functions except F10 and F13. Compared to other competing algorithms, ALCSODE exhibits superior convergence accuracy and stability on these unimodal functions. Overall, the simulation results clearly indicate that ALCSODE possesses a significant performance advantage in unimodal function optimization tasks.

2) Evaluation of multimodal banchmark: Functions F1, F6, F9, F11, F12, F14, F15, F16, and F17 are typical high-dimensional multimodal functions that contain not only a primary global optimum but also multiple local optima, making them well-suited for evaluating the exploration capability of optimization algorithms. Tables IV to VI present the optimization results of the proposed algorithm ALCSODE compared with seven other competing algorithms on these functions. Benefiting from its strong exploration ability, ALCSODE is able to locate the global optimum for functions F9, F11, F12, F14, F15, F16, and F17 after identifying the promising regions. In the optimization of F1 and F6, ALCSODE also demonstrates superior performance. The simulation results indicate that ALCSODE exhibits satisfactory exploration capability in accurately scanning the search space and avoiding local optima.

3) Evaluation of CEC 2022 benchmark: CEC2022 serves as an effective benchmark for evaluating the performance of optimization algorithms and their ability to solve complex optimization problems. It requires algorithms to handle multimodality, multi-scaled landscapes, and fused complex function shapes. The test suite consists of 12 benchmark functions, including unimodal, multimodal, hybrid, and composition functions. Specifically, F13 is a unimodal function; F14–F17 are multimodal functions; F18–F20 are hybrid functions (which may exhibit either unimodal or multimodal characteristics); and F21–F24 are composition functions with multimodal properties. Tables V and VI present the optimization results of the proposed ALCSODE algorithm compared with seven competing algorithms on the CEC2022 benchmark set. The results show that ALCSODE achieves the global optimum for F14, F15, F16, F17, and F20. For F18, F19, F21, F22, F23, and F24, ALCSODE outperforms all competitors except for F19 in the 10-dimensional case, where the global optimum is not fully reached. Overall, ALCSODE demonstrates superior optimization performance in most test cases compared to the other algorithms.

4) Statistical analysis: To provide a clear and intuitive comparison of algorithmic performance, the non-parametric Wilcoxon signed-rank test is employed to assess the statistical significance of the results. Using the best convergence value as the evaluation criterion, statistical analysis is conducted based on the average outcomes from 30 independent runs, aiming to systematically examine the effectiveness of ALCSODE. In the test outcomes, a "+" symbol indicates that ALCSODE performs better than the compared algorithm, "-" indicates relatively lower performance, and "=" denotes no significant difference. The "Rank" column represents the relative ranking in terms of solution accuracy. As shown in Table VII, ALCSODE performs competitively across most of the twelve benchmark functions. Although ALCSODE does not achieve the best performance among all algorithms on F3, F10, F12, and F13 in the 10-dimensional case, and on F10, F13, and F19 in the 50-dimensional case, slightly inferior to INFO or SHIO, it exhibits significant advantages on the remaining functions, attaining 85.5% (41 out of 48) of the best convergence results. Furthermore, the Wilcoxon rank-sum test is applied to validate the consistency of performance rankings across different dimensions. The overall results indicate that ALCSODE demonstrates strong competitiveness among all the compared algorithms and exhibits advantages in convergence accuracy from a statistical perspective.

D. Scalability and Stability Analysis of ALCSODE

This subsection analyzes the scalability and stability of ALCSODE. For scalability analysis, ALCSODE is tested on functions F1 to F24 under 10-dimensional and 50-dimensional settings, with the simulation results presented in Tables IV to VI. The analysis shows that although the performance of all algorithms declines as the dimensionality increases from 10 to 50, ALCSODE experiences a relatively smaller performance drop, demonstrating strong robustness. It continues to achieve the same optimal convergence accuracy on several functions as in the 10-dimensional case. In contrast, traditional PSO and CSO algorithms exhibit significant performance degradation in high-dimensional spaces. Other CSO variants, as well as metaheuristic algorithms such as INFO and SHIO, show moderate performance but are more sensitive to initial settings, with greater convergence fluctuations and a higher risk of premature convergence.

In terms of algorithmic stability, ALCSODE generally exhibits lower standard deviations compared to other algorithms. For instance, in both 10-dimensional and 50-dimensional experiments, ALCSODE achieves the smallest standard deviation values in most cases for functions F1, F2, F3, F4, F9, F11, F12, F13, and F21. This indicates a lower sensitivity to initial conditions and stronger resistance to random disturbances, which is particularly important for maintaining stable convergence in noisy engineering optimization environments.

V. Conclusion and Future Work

This study proposes Adaptive Levy Flight Chicken Swarm Optimization with Differential Evolution (ALCSODE) to overcome traditional CSO limitations like local optima entrapment, weak perturbation, and fixed parameters. ALCSODE enhances exploration and exploitation balance by introducing an adaptive differential perturbation in the rooster phase, combined with fitness-based step-size control and a non-inertia weight strategy. The hen phase applies dynamic scaling and Levy flights to improve local optima escape and convergence speed. An elitism preservation mechanism maintains diversity and guides solutions via information sharing. The mathematical modeling of ALCSODE and descriptions of its steps and strategies are also presented.

The performance of ALCSODE is validated on 24 benchmark functions, including unimodal, multimodal, and the

CEC2022 test suite. Results on unimodal functions demonstrate ALCSODE's strong exploitation ability, effectively converging toward the global optimum. For multimodal functions, ALCSODE exhibits robust exploration capability, successfully escaping local optima and locating the primary optimal regions. To comprehensively evaluate its effectiveness, ALCSODE is compared with seven well-known algorithms, including PSO, CSO, L-QCSO, ICSO, ECSO, SHIO, and INFO. Experimental results indicate that ALCSODE achieves superior balance between exploration and exploitation, outperforming these competitors and delivering more competitive optimization results.

However, the integration of multiple mechanisms, such as differential computation, dynamic step adjustment, and elitism preservation, may increase computational complexity, potentially affecting efficiency in large-scale or real-time optimization. Future research may focus on enhancing inter-module synergy, developing automated parameter adjustment mechanisms to reduce hyperparameter sensitivity, and exploring dimensionality reduction or decomposition strategies to improve efficiency in high-dimensional problems.

REFERENCES

[1] S. Mirjalili, "The ant lion optimizer," *Advances in engineering software*, vol. 83, pp. 80–98, 2015.

[2] R. G. Rakotonirainy and J. H. van Vuuren, "Improved metaheuristics for the two-dimensional strip packing problem," *Applied Soft Computing*, vol. 92, p. 106268, 2020.

[3] S.-H. Liu, M. Mernik, D. Hrnčič, and M. Črepinšek, "A parameter control method of evolutionary algorithms using exploration and exploitation measures with a practical application for fitting sovova's mass transfer model," *Applied Soft Computing*, vol. 13, no. 9, pp. 3792–3805, 2013.

[4] A. Lambora, K. Gupta, and K. Chopra, "Genetic algorithm-a literature review," in *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*. IEEE, 2019, pp. 380–384.

[5] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, pp. 341–359, 1997.

[6] H.-G. Beyer, "Evolution strategies," *Scholarpedia*, vol. 2, no. 8, p. 1965, 2007.

[7] S. Kirkpatrick, C. Gelatt, M. Vecchi *et al.*, "Optimization by simulated annealing. science. v220," *Lever & B. Richards (1994) The applications of generic planning architecture to ight allocation CHIC deliverable*, vol. 4, no. 3, 1983.

[8] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "Gsa: a gravitational search algorithm," *Information sciences*, vol. 179, no. 13, pp. 2232–2248, 2009.

[9] H. R. Bouchekara, "Optimal design of electromagnetic devices using a black-hole-based optimization technique," *IEEE Transactions on Magnetics*, vol. 49, no. 12, pp. 5709–5714, 2013.

[10] F. Qin, A. M. Zain, and K.-Q. Zhou, "Harmony search algorithm and related variants: A systematic review," *Swarm and Evolutionary Computation*, vol. 74, p. 101126, 2022.

[11] F. Qin, A. M. Zain, K.-Q. Zhou, and D.-B. Zhuo, "Hybrid weighted fuzzy production rule extraction utilizing modified harmony search and bpnn," *Scientific Reports*, vol. 15, no. 1, p. 11012, 2025.

[12] I. Ahmadianfar, A. A. Heidari, S. Noshadian, H. Chen, and A. H. Gandomi, "Info: An efficient optimization algorithm based on weighted mean of vectors," *Expert Systems with Applications*, vol. 195, p. 116516, 2022.

[13] H. N. Fakhouri, F. Hamad, and A. Alawamrah, "Success history intelligent optimizer," *The Journal of Supercomputing*, vol. 78, no. 5, pp. 6461–6502, 2022.

[14] A. Chakraborty and A. K. Kar, "Swarm intelligence: A review of algorithms," *Nature-inspired Computing and Optimization: Theory and Applications*, pp. 475–494, 2017.

[15] A.-Q. Tian, F.-F. Liu, and H.-X. Lv, "Snow geese algorithm: A novel migration-inspired meta-heuristic algorithm for constrained engineering optimization problems," *Applied Mathematical Modelling*, vol. 126, pp. 327–347, 2024.

[16] G. Hu, M. Cheng, E. H. Houssein, A. G. Hussien, and L. Abualigah, "Sdo: A novel sled dog-inspired optimizer for solving engineering problems," *Advanced Engineering Informatics*, vol. 62, p. 102783, 2024.

[17] A. Alkharsan and O. Ata, "Hawkfish optimization algorithm: A gender-bending approach for solving complex optimization problems," *Electronics*, vol. 14, no. 3, p. 611, 2025.

[18] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-international Conference on Neural Networks*, vol. 4. IEEE, 1995, pp. 1942–1948.

[19] M. Dorigo and L. M. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 53–66, 1997.

[20] R. Akbari, A. Mohammadi, and K. Ziarati, "A novel bee swarm optimization algorithm for numerical function optimization," *Communications in Nonlinear Science and Numerical Simulation*, vol. 15, no. 10, pp. 3142–3155, 2010.

[21] F. Pourpanah, R. Wang, C. P. Lim, X.-Z. Wang, and D. Yazdani, "A review of artificial fish swarm algorithms: Recent advances and applications," *Artificial Intelligence Review*, vol. 56, no. 3, pp. 1867–1903, 2023.

[22] X.-S. Yang, "A new metaheuristic bat-inspired algorithm (2010)," *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*, pp. 65–74, 2023.

[23] X.-S. Yang and S. Deb, "Cuckoo search via lévy flights," in *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC)*. IEEE, 2009, pp. 210–214.

[24] S. Ye, K. Zhou, A. M. Zain, F. Wang, and Y. Yusoff, "A modified harmony search algorithm and its applications in weighted fuzzy production rule extraction," *Frontiers of Information Technology & Electronic Engineering*, vol. 24, no. 11, pp. 1574–1590, 2023.

[25] A. H. Gandomi and A. H. Alavi, "Krill herd: a new bio-inspired optimization algorithm," *Communications in nonlinear science and numerical simulation*, vol. 17, no. 12, pp. 4831–4845, 2012.

[26] M. Mitić, N. Vuković, M. Petrović, and Z. Miljković, "Chaotic fruit fly optimization algorithm," *Knowledge-based systems*, vol. 89, pp. 446–458, 2015.

[27] J. James and V. O. Li, "A social spider algorithm for global optimization," *Applied Soft Computing*, vol. 30, pp. 614–627, 2015.

[28] J. Yang, L. Qu, Y. Shen, Y. Shi, S. Cheng, J. Zhao, and X. Shen, "Swarm intelligence in data science: applications, opportunities and challenges," in *International Conference on Swarm Intelligence*. Springer, 2020, pp. 3–14.

[29] J. R. Manne, "Swarm intelligence for multi-objective optimization in engineering design," in *Advanced Methodologies and Technologies in Artificial Intelligence, Computer Simulation, and Human-Computer Interaction*. IGI Global, 2019, pp. 180–194.

[30] T. Nakane, N. Bold, H. Sun, X. Lu, T. Akashi, and C. Zhang, "Application of evolutionary and swarm optimization in computer vision: a literature survey," *IPSJ Transactions on Computer Vision and Applications*, vol. 12, no. 1, p. 3, 2020.

[31] M. Xu, L. Cao, D. Lu, Z. Hu, and Y. Yue, "Application of swarm intelligence optimization algorithms in image processing: A comprehensive review of analysis, synthesis, and optimization," *Biomimetics*, vol. 8, no. 2, p. 235, 2023.

[32] S. Almufti, "Using swarm intelligence for solving nphard problems," *Academic Journal of Nawroz University*, vol. 6, no. 3, pp. 46–50, 2017.

[33] I. Fister Jr, X.-S. Yang, I. Fister, J. Brest, and D. Fister, "A brief review of nature-inspired algorithms for optimization," *ArXiv Preprint ArXiv:1307.4186*, 2013.

[34] S. Deb, X.-Z. Gao, K. Tammi, K. Kalita, and P. Mahanta, "Recent studies on chicken swarm optimization algorithm: a review (2014–2018)," *Artificial Intelligence Review*, vol. 53, pp. 1737–1765, 2020.

[35] X. Huang, C. Ye, and J. Zheng, "Chicken swarm optimization algorithm of hybrid evolutionary searching strategy," *Computer Engineering and Applications*, vol. 54, no. 7, pp. 176–181, 2018.

[36] D. Wu, F. Kong, W. Gao, Y. Shen, and Z. Ji, "Improved chicken swarm optimization," in *2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*. IEEE, 2015, pp. 681–686.

[37] X. Shi and Y. Gao, "Hybrid algorithm based on chicken swarm optimization and artificial bee colony," *Journal of Hefei University of Technology (Natural Science)*, vol. 41, no. 5, pp. 589–594, 2018.

[38] A. Meng, Z. Li, H. Yin, S. Chen, and Z. Guo, "Accelerating particle swarm optimization using crisscross search," *Information Sciences*, vol. 329, pp. 52–72, 2016.

[39] J. Zhang, K. Xia, and Z. Yin, "Quantum chicken swarm optimization with levy flight and its application in parameter optimization of random forest," in *2019 3rd International Conference on Electronic Information Technology and Computer Engineering (EITCE)*. IEEE, 2019, pp. 864–867.

[40] M. Lin, Y. Zhong, J. Lin, and X. Lin, "Enhanced chicken swarm optimisation for function optimisation problem," *International Journal of Wireless and Mobile Computing*, vol. 15, no. 3, pp. 258–269, 2018.

[41] D. Wu, S. Xu, and F. Kong, "Convergence analysis and improvement of the chicken swarm optimization algorithm," *IEEE Access*, vol. 4, pp. 9400–9412, 2016.

[42] M. Z. Afzal, F. Wen, N. Saeed, and M. Aurangzeb, "Enhanced state of charge estimation in electric vehicle batteries using chicken swarm optimization with open ended learning," *Scientific Reports*, vol. 15, no. 1, p. 10833, 2025.

[43] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67–82, 1997.

[44] S. Das and P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 4–31, 2010.

[45] X. Meng, Y. Liu, X. Gao, and H. Zhang, "A new bio-inspired algorithm: chicken swarm optimization," in *Advances in Swarm Intelligence: 5th International Conference, ICSI 2014, Hefei, China, October 17-20, 2014, Proceedings, Part I 5*. Springer, 2014, pp. 86–94.

[46] S. Deb, X.-Z. Gao, K. Tammi, K. Kalita, and P. Mahanta, "Recent studies on chicken swarm optimization algorithm: a review (2014–2018)," *Artificial Intelligence Review*, vol. 53, pp. 1737–1765, 2020.

[47] D. Yazdani, J. Branke, M. N. Omidvar, X. Li, C. Li, M. Mavrovouniotis, T. T. Nguyen, S. Yang, and X. Yao, "Ieee cec 2022 competition on dynamic optimization problems generated by generalized moving peaks benchmark," *arXiv preprint arXiv:2106.06174*, 2021.

[48] T. M. Shami, A. A. El-Saleh, M. Alswaitti, Q. Al-Tashi, M. A. Summakieh, and S. Mirjalili, "Particle swarm optimization: A comprehensive survey," *IEEE Access*, vol. 10, pp. 10 031–10 061, 2022.

[49] F. Han, Q. Zhao, Z. Du *et al.*, "Enhanced chicken swarm algorithm for global optimization," *Application Research of Computers*, vol. 36, no. 8, pp. 2317–2319, 2019.

[50] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *MHS'95. Proceedings of the sixth international symposium on micro machine and human science*. IEEE, 1995, pp. 39–43.

Public Opinion Stage Segmentation in Disaster Events: A Study Based on Multimodal Sentiment Prediction Model

Xiaogang Yuan, Jiaxi Chen, Dezhi An, Xiang Gong
School of Cyber Security, Gansu University of Political Science and Law, Lanzhou 730070, China

***Abstract*—Existing approaches for predicting public sentiment and analyzing opinion evolution during disaster events using multimodal data (text, video, audio) suffer from several limitations: an inadequate dynamic fusion of heterogeneous multi-source data, and imprecise division of public opinion stages. To address these issues, this paper proposes an Enhanced Disentangled Cross Fusion (EDCF) model-based framework for analyzing the evolution of public opinion in disaster events. This framework integrates the E-Divisive with Medians (EDM) change point detection method with spatiotemporal sequence modeling techniques to achieve fine-grained stage segmentation. The EDCF model employs Transformers and positional encoding to process time-series signals (audio/video), effectively capturing long-range dependencies. It enhances modality-specific representation capabilities by introducing dedicated encoders for each modality, a shared encoder, and a reconstruction decoder for disentangled representation learning. Furthermore, the model utilizes a cross-modal language-guided attention mechanism for efficient and effective feature fusion. Experimental validation on the publicly available multimodal sentiment dataset CMU-MOSI demonstrates that the proposed EDCF framework significantly outperforms baseline methods on key sentiment prediction metrics.**

***Keywords*—Multimodal sentiment prediction; e-divisive with medians; transformer; encoder; decoder; cross-attention**

I. Introduction

Disaster events (e.g., earthquakes, and floods) often trigger intense fluctuations in public sentiment. Related multimodal content (text, video, audio) rapidly disseminates through social media platforms, where the evolution of public opinion directly influences public behavior and government emergency decision-making. Consequently, accurately segmenting and analyzing the evolution of public opinion is crucial for enhancing the effectiveness of emergency responses.

A. Multimodal Sentiment Analysis

Multimodal Sentiment Analysis (MSA) aims to improve the accuracy and comprehensiveness of sentiment recognition by fusing heterogeneous information from multiple sources such as text, audio, and video. Early research primarily employed feature concatenation methods for simplistic cross-modal fusion. Two common fusion strategies are: early fusion and late fusion. Early fusion constructs a joint feature representation by extracting features from each modality and merging them at the input layer [1], [2], [3], [34]. Late fusion first performs sentiment analysis on each modality individually and then integrates the unimodal sentiment decisions into a final decision using various mechanisms [4], [5], [6], [35]. However, these methods often introduce redundancy and conflicts due to insufficient consideration of semantic differences and interaction relationships between modalities.

Recent research has shifted towards deep learning-driven cross-modal modeling, designing more sophisticated fusion strategies to enhance performance. Attention has moved beyond simple fusion to explicitly modeling inter-modal interactions. For instance, Zadeh et al. [7] proposed a tensor fusion network that learns both intra-modal and cross-modal dynamics end-to-end for the three modalities, aiming to improve MSA performance. Han et al. [8] introduced a method that simultaneously maximizes mutual information (MI) between modalities and between the multimodal fusion result and the unimodal inputs, thereby enhancing model capability. Furthermore, contrastive learning frameworks have significantly improved MSA accuracy by optimizing semantic representations through cross-modal alignment. Research on contrastive learning can be categorized into two groups: self-supervised contrastive learning [9], [10], [11], [36], [37] and supervised contrastive learning [12], [13], [38], [39]. The key distinction lies in whether label information is used to guide the selection of positive and negative sample pairs.

Recent studies further focus on modality noise handling and multimodal alignment techniques. For example, Li et al. [14] proposed an improved FGM-CM-BERT model. It optimizes adversarial training using the Fast Gradient Method (FGM) with an adaptive parameter adjustment strategy to enhance model generalization. Combined with a multi-head attention mechanism for efficient cross-modal feature fusion, experiments showed it achieved 48.2% accuracy for seven-class emotion classification on the CMU-MOSI dataset. Wang et al. [15] proposed the DLF model, which enhances the accuracy and efficiency of multimodal data fusion by introducing components such as a feature decomposition module, a language-guided feature attractor (LFA), and hierarchical prediction. Despite significant progress, effectively balancing semantic differences and dynamic interactive relationships between modalities remains a core challenge in MSA.

B. Public Opinion Stage Segmentation

Public opinion stage segmentation aims to identify the lifecycle of opinion propagation, providing a theoretical basis for dynamic monitoring and intervention. Early research often relied on propagation dynamics models (e.g., SIR model, [40], [41]) or time-series analysis to model the opinion dissemination process. For example, Zeng et al. [16] proposed

segmenting opinion propagation into stages: germination, outbreak, diffusion, climax, and subsidence, combined with the SIR model to quantify propagation intensity at each stage. However, such methods are typically limited to a single text modality, struggling to capture the complex evolution patterns of multimodal public opinion.

Recent studies attempt to segment stages based on trough detection in propagation volume curves. For instance, Chen et al. [17] employed the Fisher optimal segmentation algorithm to cluster daily short-video public opinion data, roughly dividing online opinion into starting, outbreak, decline, and subsidence stages. Wei et al. [18] refined the lifecycle into four stages (development period, primary outbreak period, secondary outbreak period, and decline period) based on repost volume. Although these methods improve granularity and interpretability in stage segmentation, their reliance on propagation volume metrics makes them susceptible to noise. Furthermore, they fail to adequately incorporate sentiment dimensions and user behavioral characteristics, limiting their practical effectiveness.

C. Summary and Research Challenges

In summary, while significant advancements have been made in both the techniques and methodologies of multimodal sentiment analysis and public opinion evolution research, several key challenges persist:

- Multimodal Sentiment Analysis: Effectively balancing semantic differences and dynamic interactive relationships between modalities remains a core challenge.
- Opinion Stage Segmentation: Existing segmentation methods lack sufficient dynamism and fail to adequately integrate sentiment dimension features, limiting their practical applicability.

To address these challenges, the main contributions of this study are summarized as follows:

- We optimize the DLF model proposed by Wang et al. [15], constructing a superior Enhanced Disentangled Cross Fusion (EDCF) model. This model effectively tackles three key challenges in multimodal sentiment prediction: inter-modal asynchrony, noisy modality interference, and the complex modeling of cross-modal interactions.
- We integrate E-Divisive with Medians (EDM) change point detection with the sentiment features captured by the EDCF model to achieve finer-grained segmentation of public opinion stages.
- The collaborative modeling approach, integrating multimodal sentiment analysis with public opinion phase division based on EDM anomaly detection, enhances the accuracy of public opinion monitoring. This facilitates real-time online surveillance and provides early warning mechanisms for relevant authorities, enabling timely intervention and guidance to prevent the recurrence of negative events such as cyber violence.

The remainder of this paper is structured as follows. Section II details the construction of the proposed EDCF multimodal sentiment model. Section III applies the EDCF model combined with the EDM change point detection method to achieve precise segmentation of public opinion stages for specific disaster events. Section IV presents experimental validation demonstrating that the proposed EDCF model significantly outperforms traditional fusion methods in cross-modal interaction and fusion efficiency. This section also validates the effectiveness and necessity of the proposed opinion stage segmentation method. Finally, Section V concludes the study and suggests directions for future research.

II. Multimodal Sentiment Prediction Model

This paper proposes Enhanced Disentangled Cross Fusion (EDCF) model for multimodal sentiment prediction, based on dual latent feature learning and cross-modal attention interaction. The core architecture of the proposed model (Fig. 1) comprises four key modules:

- Enhanced Multimodal Feature Extraction Module: Employs Transformers and positional encoding to process time-series signals (audio/video), effectively capturing long-range dependencies.
- Dual-Stream Feature Disentanglement and Reconstruction Module: Enhances the representational capacity of each modality by introducing dedicated encoders (per modality), a shared encoder, and a reconstruction decoder for disentangled representation learning.
- Hierarchical Cross-Modal Attention Interaction Module: Achieves efficient feature fusion through a language-guided cross-attention mechanism.
- Dynamic Feature Fusion Module: Innovatively incorporates a differentiable weighting strategy to integrate modality-specific features and cross-modality interactive features effectively.

A. Multimodal Feature Extraction and Enhancement Module

Initial independent feature extraction is performed for each modality:

- Text Modality: The pre-trained BERT model https://huggingface.co/google-bert/bert-base-uncased is used for tokenization, generating word vectors of dimension (50, 768), denoted as $I_l \in \mathbb{R}^{T_l \times d_l}$.
- Audio Modality: Fundamental acoustic features are extracted using the Librosa library, including Zero Crossing Rate (ZCR, 1 dimension), Mel-frequency cepstral coefficients (MFCC, 20 dimensions), and Chroma features based on the Constant-Q Transform (Chroma CQT, 12 dimensions), resulting in a total of 33 dimensions. To ensure consistent sequence length, all audio data is uniformly truncated or zero-padded to a fixed length of 50 frames. Subsequently, Principal Component Analysis (PCA) is applied to compress the 33-dimensional features into 5 dimensions, preserving essential information. The processed audio representation has a dimension of (50, 5), denoted as $I_a \in \mathbb{R}^{T_a \times d_a}$.
- Video Modality: OpenCV's Hear cascade classifier extracts 50 frames containing faces from the video.

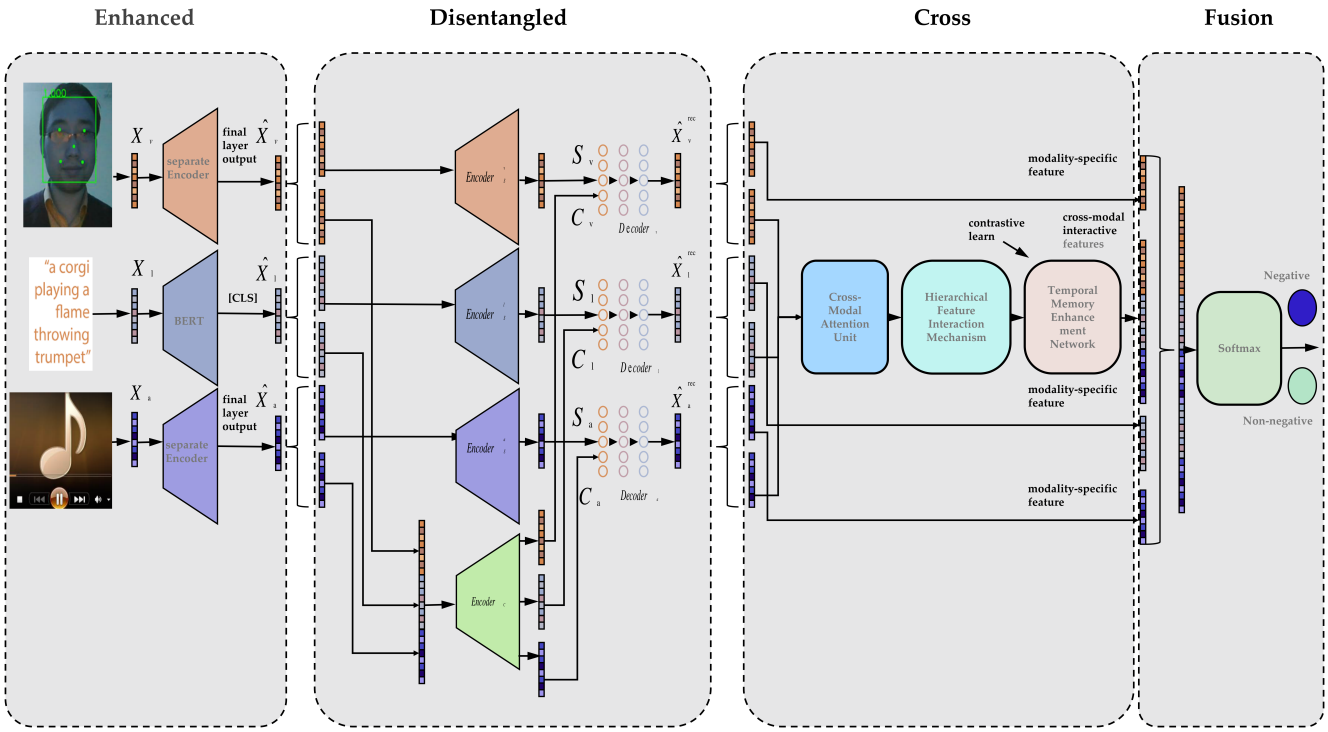


Fig. 1. Structural diagram of the EDCF (Enhanced Disentangled Cross Fusion) model.

For each frame, facial features are extracted using the Google-developed FaceNet model. PCA is similarly applied to compress the original 128-dimensional features into 20 dimensions, retaining critical information. The final video representation has a dimension of (50, 20), denoted as $I_v \in \mathbb{R}^{T_v \times d_v}$. Here, $\mathrm{T}_{\mathrm{m}\in\{\mathrm{l,a,v}\}}$ represents the sequence length common to all modalities, and $\mathrm{d}_{\mathrm{m}\in\{\mathrm{l,a,v}\}}$ represent the corresponding feature vector dimensions.

To enhance feature extraction capability and improve model robustness, a noise-resistant common vector representation is generated for each modality:

- For the text modality, the feature I_l is directly fed into the pre-trained BERT model (https://huggingface.co/google-bert/bert-base-uncased), and the output vector of the [CLS] token is used as its common representation $I_l^c \in \mathbb{R}^{1\times d_l}$.
- For the video and audio modalities, the features I_v and I_a are input into separate Transformer encoders. The last vector from the final layer output of each encoder is taken as their respective common representations, denoted as $I_v^c \in \mathbb{R}^{1\times d_v}$ and $I_a^c \in \mathbb{R}^{1\times d_a}$. This processing effectively mitigates noise interference.

The core objective of modality alignment is to project the common representations of different modalities into a shared semantic space. First, one-dimensional convolutional networks (Conv1D) are used to transform the three common vectors I_l^c, I_v^c, I_a^c into the same dimension d_c, yielding $L_c \in \mathbb{R}^{d_c\times 1}$, $V_c \in \mathbb{R}^{d_c\times 1}$, and $A_c \in \mathbb{R}^{d_c\times 1}$. Subsequently, a projection layer maps these features into a unified latent space [as shown in Eq. (1)].

$$\widehat{\mathrm{X}}_{\mathrm{m}} = \mathrm{Conv1D}(\mathrm{X}_{\mathrm{m}}), \mathrm{m} \in \{\mathrm{l, a, v}\} \tag{1}$$

B. Dual-Stream Feature Decoupling and Reconstruction Module

This module explicitly separates Modality-specific Features from Cross-modal Shared Features through parallel parameter-isolated encoders and parameter-shared encoders. A triple-constraint mechanism ensures robustness during the decoupling process. Its core architecture is shown in Fig. 2.

The modality-specific encoder captures the unique characteristics of each modality using a parameter-isolated Transformer network. Given the input features $\widehat{X}_m \in R^{T_m\times d_s}$ for modality m, its processing is defined by Eq. (2). Here, d_s is the dimension of the specific features, and $Encoder_s^m$ contains Transformer modules with independent layer parameters.

The cross-modal shared encoder extracts cross-modal common representations using a parameter-shared Transformer network, as defined by Eq. (3). Here, d_c is the shared feature dimension, $Encoder_c$ forcibly reused across modalities to promote consistent mapping of different modal data in the latent space. The shared feature dimension is dynamically adjusted via a learnable scaling factor, enabling adaptive alloca- tion of feature capacity.

$$\mathrm{S}_{\mathrm{m}} = \mathrm{Encoder}_{\mathrm{s}}^{\mathrm{m}}(\widehat{\mathrm{X}}_{\mathrm{m}}) \in \mathrm{R}^{\mathrm{T_m}\times \mathrm{d_s}} \tag{2}$$

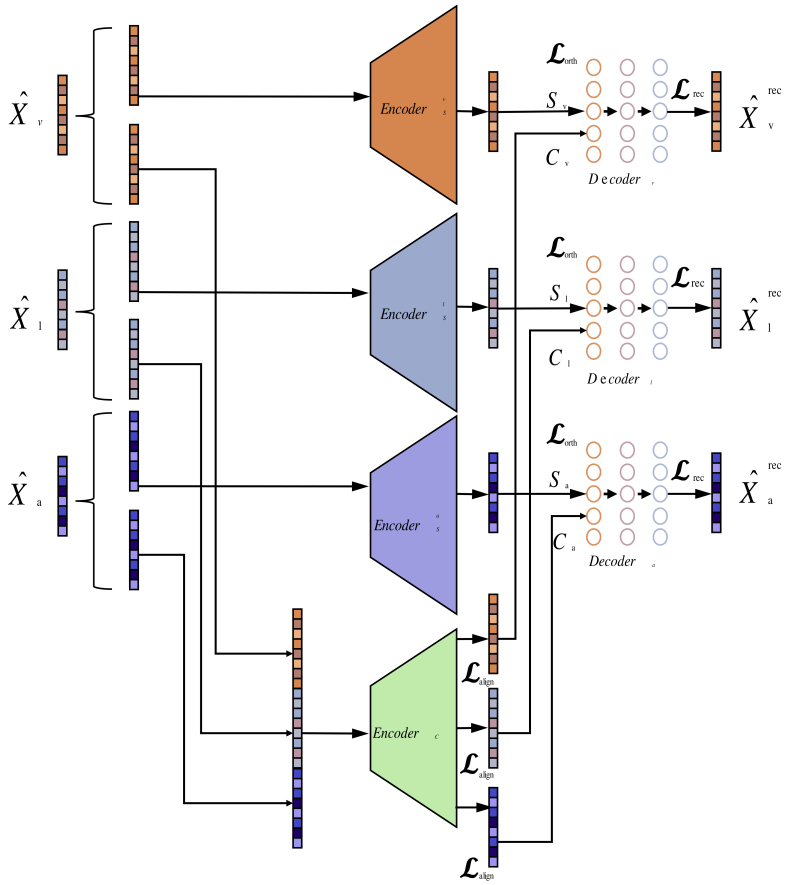


Fig. 2. Core architecture diagram of the dual-stream feature decoupling and reconstruction module.

$$\mathbf{C}_m = \text{Encoder}_c(\hat{\mathbf{X}}_m) \in \mathbb{R}^{T_m \times d_c} \tag{3}$$

To validate the effectiveness of feature decoupling, a cross-modal feature reconstruction task is designed. First, the specific features of modality and the shared features of modality are concatenated and input into a decoder, as shown in Eq. (4). Here, $\oplus$ denotes channel-wise concatenation, and the decoder is implemented using 1D transposed convolutional layers. Then, reconstruction loss is minimized to ensure information integrity, as shown in Eq. (5).

$$\hat{X}_m^{rec} = \text{Decoder}_m([S_m \oplus C_m]) \in \mathbb{R}^{T_m \times d} \tag{4}$$

$$\mathcal{L}_{rec} = \sum_m \|\hat{X}_m - \hat{X}_m^{rec}\|_F^2 \tag{5}$$

To enhance the robustness of feature decoupling, dual regularization constraints are constructed:

- Spatial Orthogonality Constraint: Forces modality-specific features and shared features to be orthogonal in the feature space, minimizing their spatial correlation, as specified in Eq. (6).
- Similarity Alignment Constraint: Requires cross-modal shared features to satisfy distribution consistency, as specified in Eq. (7).

$$\mathcal{L}_{orth} = \sum_m \frac{1}{T_m} \sum_{t=1}^{T_m} |S_m^{(t)} \cdot C_m^{(t)}| \tag{6}$$

$$\mathcal{L}_{align} = \sum_{m \neq n} \text{KL}(\text{Align}(C_m) \| \text{Align}(C_n)) \tag{7}$$

The collaborative design of parameter isolation (specific encoder) and parameter sharing (shared encoder), coupled with the learnable dimension scaling factor $\alpha = d_s/(d_s + d_c)$ for adaptive adjustment of the capacity ratio between the two types of features, achieves fine-grained feature decoupling. The joint optimization of $\mathcal{L}_{\text{rec}}$, $\mathcal{L}_{\text{orth}}$, and $\mathcal{L}_{\text{align}}$ ensures the interpretability of the decoupling process.

C. Cross-Modal Hierarchical Attention Interaction Module

This module implements cross-modal feature interaction and temporal dependency modeling through a multi-granularity attention mechanism. Its hierarchical structure is shown in Fig. 3. The module contains three core components: Cross-Modal Attention Unit, Temporal Memory Enhancement Network, and Hierarchical Feature Interaction Mechanism.

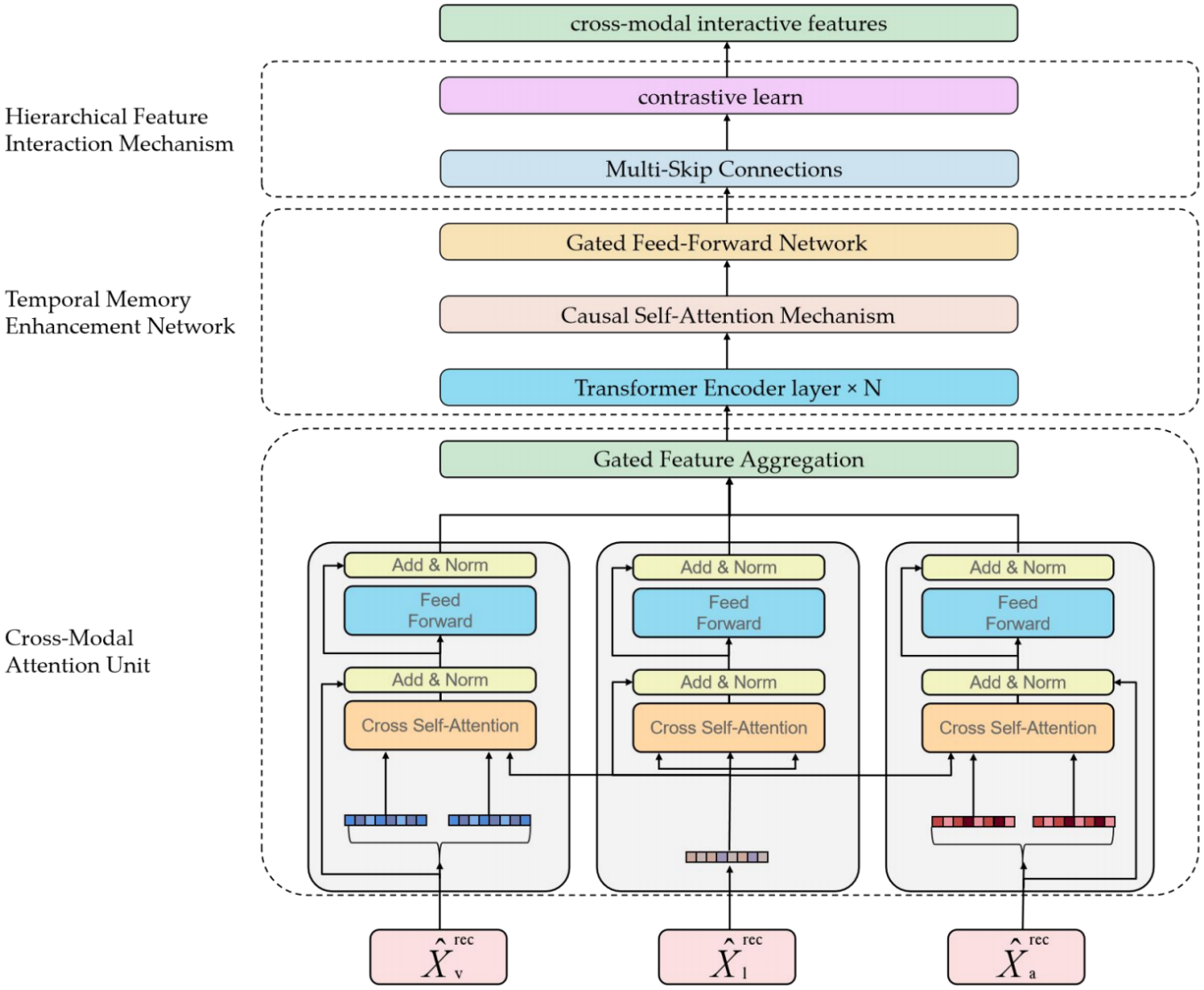


Fig. 3. Schematic diagram of the hierarchical structure for the cross-modal hierarchical-attention interaction module.

The Cross-Modal Attention Unit achieves fine-grained feature alignment based on a bidirectional cross-attention unit. Its computation process is shown in Eq. (8) and (9). Here, denotes Multi-Head Attention, calculated as shown in Eq. (10). Gated feature aggregation introduces adaptive weights to fuse contributions from different modalities, as shown in Eq. (11) and Eq. (12). Here, σ is the Sigmoid function, W_g^{la} is the gating parameter matrix, enabling adaptive fusion of cross-modal features through dynamic weight allocation.

$$H_{l\leftarrow a} = \text{CrossAttn}(Q = S_l, K = S_a, V = S_a) \in \mathbb{R}^{T_l \times d} \quad (8)$$

$$H_{l\leftarrow v} = \text{CrossAttn}(Q = S_l, K = S_v, V = S_v) \in \mathbb{R}^{T_l \times d} \quad (9)$$

$$\text{CrossAttn}(Q, K, V) = \text{Softmax}\left(\frac{QW_Q(KW_K)^T}{\sqrt{d}}\right)VW_V \quad (10)$$

$$g_{la} = \sigma(W_g^{la}[S_l, H_{l\leftarrow a}]) \quad (11)$$

$$H_l^{cross} = g_{la} \odot H_{l\leftarrow a} + (1 - g_{la}) \odot H_{l\leftarrow v} \quad (12)$$

The Temporal Memory Enhancement Network employs a deep Transformer Encoder to construct a temporal memory enhancement network. Its hierarchical processing is shown in Eq. (13). Here, $MemEnc_l$ contains L_m Transformer layers, each comprising two core components:

- Causal Self-Attention Mechanism: Ensures temporal directionality via a mask matrix, preventing future information leakage, as shown in Eq. (14).

- Gated Feed-Forward Network: Employs dual nonlinear projections to enhance temporal pattern capture capability, as shown in Eq. (15).

$$H_l^{mem} = \text{MemEnc}_l(\text{LayerNorm}(H_l^{cross} + S_l)) \tag{13}$$

$$\text{CausalAttn(X)} = \text{Tril}(\text{QK}^{\text{T}}/\sqrt{\text{d}})\text{V} \tag{14}$$

$$GatedFFN(X) = (GeLU(XW_1) \odot XW_2)W_3 \tag{15}$$

The Hierarchical Feature Interaction Mechanism constructs multi-level representations via Multi-skip connections, mathematically expressed in Eq. (16). Here, α_k is a learnable scaling weight, and $Downsample_k$ denotes average pooling with stride 2^{k-1}. Contrastive learning is utilized to optimize cross-modal consistency, bringing cross-modal sample representations closer in the semantic space, as shown in Eq. (17). Here, h_m^i is the high-level feature of instance i for a modality m, and τ is the temperature coefficient. Theoretical analysis indicates that when the condition in Eq. (18) is satisfied, the hierarchical attention mechanism effectively enhances cross-modal interaction efficiency. This gradient propagation form ensures high-level semantic features gradually align during contrastive learning.

$$\text{H}_\text{l}^{\text{final}} = \sum_{\text{k=1}}^{\text{K}} \alpha_\text{k} \cdot \text{Downsample}_\text{k}(\text{H}_\text{l}^{\text{mem,(k)}}) \tag{16}$$

$$\mathcal{L}_{\text{sem}} = -\log \frac{\sum \exp\left(\cos\left(h_1^i, h_a^i\right)/\tau\right)}{\sum \exp\left(\cos\left(h_1^i, h_a^j\right)/\tau\right)} \tag{17}$$

$$\frac{\partial \mathcal{L}_{\text{sem}}}{\partial \mathcal{H}_\text{m}} \propto \text{E}[\cos(\text{H}_\text{m}, \text{H}_\text{n})] \cdot \frac{\partial \cos(\text{H}_\text{m}, \text{H}_\text{n})}{\partial \text{H}_\text{m}} \tag{18}$$

D. Dynamic Feature Fusion Module

This module achieves a dynamic fusion of multi-level features through an adaptive gating mechanism, effectively integrating modality-specific features and cross-modal interactive features.

Within the same modality, dynamic gated fusion is performed using modality-aware attention to compute dynamic weights. This process is formalized by Eq. (19) and (20), where $W_g^l \in \mathbb{R}^{3d_h \times 3}$ is a learnable parameter matrix ensuring weight normalization. This gating mechanism also incorporates a differential entropy constraint via a regularization term to prevent weight collapse, as shown in Eq. (21).

$$\text{g}_\text{l} = \text{Softmax}\left(\frac{[\text{z}_\text{l}^{\text{spec}}; \text{z}_\text{l}^{\text{cross}}; \text{z}_\text{l}^{\text{share}}]\text{W}_\text{g}^\text{l}}{\sqrt{\text{d}_h}}\right) \in \mathbb{R}^3 \tag{19}$$

$$\widehat{\text{H}}_\text{l} = \sum_{\text{k=1}}^{3} \text{g}_\text{l}^{\text{(k)}} \cdot [\text{z}_\text{l}^{\text{spec}}, \text{z}_\text{l}^{\text{cross}}, \text{z}_\text{l}^{share}]_\text{k} \tag{20}$$

$$\mathcal{L}_{\text{ent}} = -\frac{1}{3}\sum_{\text{m}}\sum_{\text{k=1}}^{3} \text{g}_\text{l}^{\text{(k)}} \log \text{g}_\text{l}^{\text{(k)}} \tag{21}$$

For fusion between different modalities, a collaborative gating mechanism is employed, formalized by Eq. (22), where σ is the Sigmoid function and $W_\beta \in \mathbb{R}^{3d_h \times 3}$ is the parameter matrix. The final fused feature is given by Eq. (23). This gating mechanism incorporates multi-granularity residual learning to enhance model robustness against minor feature variations.

$$\beta = \sigma\left(\text{W}_\beta[\widehat{\text{H}}_\text{l}; \widehat{\text{H}}_\text{a}; \widehat{\text{H}}_\text{v}]\right) \in \mathbb{R}^3 \tag{22}$$

$$\text{H}_{\text{fusion}} = \beta_\text{l}\widehat{\text{H}}_\text{l} + \beta_\text{a}\widehat{\text{H}}_\text{a} + \beta_\text{v}\widehat{\text{H}}_\text{v} \in \mathbb{R}^{\text{d}_h} \tag{23}$$

III. Research Methodology Based on Multimodal Sentiment Prediction Model and EDM Anomaly Detection for Public Opinion Phase Division

A. Robustness Testing of the Multimodal Sentiment Prediction Model

To verify the robustness of the proposed multimodal sentiment prediction model EDCF (Section II) on disaster datasets, this section trained it using specific multimodal data from disaster-related emergencies. First, Python was used to crawl multimodal objects (including video, audio, text, and corresponding IDs) for short videos related to the disaster public opinion event "Yunlinkou Landslide in Sichuan" from the Douyin platform. After data preprocessing, 608 valid data entries were obtained.

Subsequently, each multimodal object was annotated into two categories: negative and non-negative. This experiment invited two postgraduate students, whose research focuses primarily on language and semantic recognition. Annotations were formally included in the database only when the results were consistent. For inconsistent annotations, an expert holding a psychological counseling certificate was invited for secondary annotation. The majority vote (2:1) determined the final sentiment polarity. Among the 608 annotated valid data entries, 417 were labeled as non-negative sentiment and 191 as negative sentiment. Of these, 58.48

After multiple tests comparing manual annotations with intelligent classification results, the multimodal sentiment prediction model EDCF proposed in Chapter 3 achieved a sentiment binary classification accuracy consistently exceeding 74.6

B. EDM Change-Point Detection Technology

EDM (E-Divisive with Medians) is a non-parametric time series and data stream change-point detection technique. Its innovation lies in accurately locating change points in data distribution structures without presupposing a data distribution model. Compared to traditional parametric methods, EDM effectively overcomes the sensitivity of conventional mean-based metrics to outliers by constructing a measurement system based on median values in multidimensional data feature spaces. The core theoretical foundation of this method is the innovative application of the Energy Distance statistic. Its technical implementation path comprises two key steps:

- Quantitative Assessment Based on Statistical Distance Between Subsequences: Construct a metric for distribution difference; when this metric exceeds a preset threshold, a potential change point is identified.
- Using Median Instead of Mean as the Difference Metric Benchmark: This design significantly enhances the algorithm's robustness against outliers.

Experimental studies show that the EDM method is not only suitable for high-dimensional data analysis scenarios but also maintains stable detection performance even in the presence of noise interference or data contamination, demonstrating strong environmental adaptability.

C. Mathematical Modeling of EDM Change-Point Detection Technology

- Define Multimodal Sentiment Statistic: This statistic aims to establish a quantitative representation of public opinion intensity. By setting a time window length ($\tau = 2$ hours) for integral statistics, unstructured multimodal information (video content) is transformed into structured time series signals. Specifically, the number of negative short videos within the k-th time window is defined as: $N_k = \sum_{t\in\mathcal{T}_k} \mathbb{I}(s_t < 0.5)$, where $s_t \in \{0, 1\}$ represents the sentiment prediction score of the i-th short video (1 for negative), $\mathcal{T}_k$ is the index set for the k-th time window, and $\mathbb{I}(\cdot)$ is the indicator function. Using the indicator function for sentiment discretization effectively avoids the cumulative effect of continuous prediction errors. The selection of a $\tau = 2$hour time window balances event response immediacy and statistical significance requirements.
- Set Change-Point Detection Energy Function: This function constructs a metric for heterogeneity in time series data. The parameter min_size specifies the minimum length of each segment after splitting, ensuring statistically significant segmentation results and avoiding unreliable splits due to overly short data.For a candidate split point i, the energy statistic between the preceding segment data $\bar{D}_1^i$ and the succeeding segment data $\bar{D}_{i+1}^n$ is shown in Eq. (24). Here, $\bar{D}_1^i = \frac{i}{\sum_{k=1}^{i} N_k}$, $\bar{D}_{i+1}^n = \frac{n-i}{\sum_{k=i+1}^{n} N_k}$. This function introduces a weighting factor $\frac{i(n-i)}{n}$ to suppress the tendency to split near the sequence endpoints. The L^2 norm based on mean difference effectively captures abrupt changes in distribution parameters and is sensitive to mean-shift type change points.

$$\mathrm{E(i;D)} = \frac{\mathrm{i(n-i)}}{\mathrm{n}} ||\bar{\mathrm{D}}_1^{\mathrm{i}} - \bar{\mathrm{D}}_{\mathrm{i+1}}^{\mathrm{n}}||_2^2 \tag{24}$$

- Set Regularization Penalty Term: This penalty term controls model complexity and over-segmentation risk. Its design reflects the regulatory role of key parameters, as shown in Eq. (25). Here, δ is the degree parameter, and n is the length of the current data segment. degree controls the calculation method of the penalty term, influencing the preference for split point locations. When $\delta = 1$, the penalty term is $\beta \times n^\alpha$, independent of the split location, applying the same penalty to all candidate points. When $\delta = 2$, the penalty term is $\beta \times n^\alpha \times \left(\frac{i}{n}\right)^2$, where the split location proportion n is $\frac{i}{n}$. $\delta = 2$ constructs a "middle suppression" effect via the quadratic term, aligning with the characteristic of public opinion events erupting at the beginning and end; β is the strength coefficient for the penalty term, balancing model likelihood and complexity through the Lagrange multiplier principle, controlling segmentation sparsity. Increasing strengthens the penalty term, reduces the adjusted energy value, and consequently decreases the number of detected change points, suppressing over-segmentation; α regulates the growth rate of the penalty term with segment length. The penalty term is proportional to n^α. Larger α imposes heavier penalties on long segments, avoiding frequent segmentation in long sequences.

$$P(i; n, \beta, \alpha, \delta) = \begin{cases} \beta n^\alpha & \text{if } \delta = 1 \\ \beta n^\alpha \left(\dfrac{i}{n}\right)^2 & \text{if } \delta = 2 \end{cases} \tag{25}$$

- Set Optimization Objective Function: This function aims to achieve optimal segmentation under statistical significance constraints, using the adjusted maximum energy criterion shown in Eq. (26). The constraint $n_{\min} = 12$ ensures the minimum analysis unit (equivalent to 24 hours of data), conforming to the public opinion event lifecycle theory. The penalty term here acts as a regularizer, generalizing the Bayesian Information Criterion (BIC) to time-varying scenarios.

$$\hat{\mathrm{i}} = \arg \max_{\mathrm{i}\in[\mathrm{n}_{\min}, \mathrm{n}-\mathrm{n}_{\min}]} [\mathrm{E(i;D)} - \mathrm{P(i;n,}\beta, \alpha, \delta)] \tag{26}$$

- Set Recursive Segmentation Criterion: This criterion constructs a multi-level framework for event evolution analysis, adopting a median strategy for iterative segmentation, as shown in Eq. (27). Segmentation significance requires: $E(\hat{i}; \mathcal{D}) > C_\alpha \cdot \sigma_N^2$, where C_α is the significance level threshold and σ_N^2 is the global variance estimate. The significance test condition sets an adaptive threshold based on extreme value theory to avoid spurious segmentation.

$$\mathcal{S}(\mathcal{D}) = \begin{cases} \emptyset & if\ n < 2n_{min} \\ \{\hat{i}\}\ U\ S(D_1^{\hat{i}})\ U\ S(D_{\hat{i}+1}^n) & otherwise \end{cases} \tag{27}$$

This mathematical model balances detection sensitivity and robustness through parameter combinations ($n_{\min}$, β, α, δ): When experimental data is short or has many change points, min_size, β, and α should be reduced, using $\delta = 1$. When data is long or has few change points, min_size, β, and α should be increased, using $\delta = 2$ to suppress middle segmentation. The theoretical basis for its parameter setting can be summarized as Eq. (28).

$$\beta = 0.008 \propto \frac{\log n}{n^\alpha}, \alpha = 1 \Rightarrow \text{LinearPenaltyGrow}th \tag{28}$$

Essentially, this framework constructs a Regularized Maximum Likelihood Estimator (Regularized MLE). Its parameter settings can be derived based on the Wiener filter theory to achieve a Pareto optimal balance between detection sensitivity and false alarm probability. Given that the public opinion evolution data analyzed in this experiment is of medium length, the final balanced parameter settings adopted were: $\min_size = 12$, $\beta = 0.008$, $\alpha = 1$, $\delta = 1$.

D. Application of EDM Change-Point Detection Technology in Public Opinion Phase Division

This study combines the multimodal sentiment prediction model proposed in Section II with the EDM change-point detection technology to divide the public opinion phases for a specific event. The specific process is as follows:

- Sentiment Prediction and Statistics: Utilize the trained multimodal sentiment prediction model from Chapter 3 to perform binary sentiment classification prediction (output: negative or non-negative sentiment) for each multimodal short video. Subsequently, count the number of short videos with negative sentiment within each time window.
- Time Series Waveform Construction: Plot a waveform graph of the number of negative videos per time slice, with time as the x-axis and the number of negative sentiment short videos per time period as the y-axis.
- Parameter Setting: Manually set relevant parameters of the EDM algorithm according to specific requirements to balance computational efficiency and detection accuracy.
- Change-Point Detection and Phase Division: Apply the EDM change-point detection technology to effectively identify abrupt change points in complex time series data based on energy statistics and media strategy. Finally, divide the different phases of public opinion development based on the detected change points and changes in the amplitude of the time slice negative video count waveform.

E. Results of Public Opinion Phase Division Using EDM Change-Point Detection

The EDM change-point detection method was applied to monitor the public opinion evolution of the "2023 Beijing-Tianjin-Hebei Rainstorm Event", identifying 11 structural change points (as shown in Fig. 4). Based on these change points and the waveform amplitude characteristics of the negative sentiment short video count within time slices, the public opinion event was divided into five key intervals. The average hourly number of negative sentiment short videos in each interval was 2.94, 4.82, 8.40, 1.90, and 0.09, respectively (details in Table I).

Based on public opinion lifecycle theory and through in-depth analysis of the above quantitative characteristics, this study conducted a structured analysis of the event's public opinion phases. The results indicate that these five intervals exhibit significantly distinct characteristics, corresponding to the five typical stages of public opinion evolution: Latent Period, First Outbreak Period, Second Outbreak Period, Decline Period, and Extinction Period.

TABLE I. INTERVALS IDENTIFIED USING THE EDM CHANGE-POINT DETECTION METHOD

No.	Interval	Start Point	End Point	Average
1	(0,36)	2023/07/28 06:00	2023/07/29 18:00	2.94
2	(37,114)	2023/07/29 19:00	2023/08/02 00:00	4.82
3	(115,164)	2023/08/02 01:00	2023/08/04 02:00	8.40
4	(165,308)	2023/08/04 03:00	2023/08/10 02:00	1.90
5	(309,+∞)	2023/08/10 03:00	+∞	0.09

IV. EXPERIMENTAL RESULTS

A. Experimental Results of Multimodal Sentiment Analysis Model

1) Training dataset: To train the sentiment analysis model and validate its effectiveness, this study employs the preprocessed CMU-MOSI dataset [19]. CMU-MOSI is a multimodal opinion-level sentiment intensity corpus containing transcribed speech, visual postures, and automatically extracted audio and visual features. Its key strengths are high inter-coder agreement in sentiment intensity annotations and precise alignment among text, video, and audio features. Sentiment intensity is defined on a linear scale ranging from -3 (strongly negative) to +3 (strongly positive), divided into seven levels, with 0 representing neutrality. Positive sentiment intensity increases progressively from -3 to 3. Annotations were completed by Amazon Mechanical Turk workers with ¿95% approval rates, with each video segment independently scored by five annotators; the average score serves as the final sentiment intensity label. The CMU-MOSI dataset provides rich and precise data support for multimodal sentiment prediction research.

Given the large memory footprint of the raw data, the practical CMU-MOSI dataset contains feature vectors for text, video, and audio modalities from 2,199 short multimodal video segments. The dataset is split as follows: 1,284 samples for training, 229 for validation, and 686 for testing. The dimensionalities of each modality's feature vectors are: text: (50, 768); video: (50, 20); audio: (375, 5).

2) Evaluation metrics and parameters: To ensure fair evaluation, this study reports model performance on both regression and classification tasks. For regression, we report Mean Absolute Error (MAE) and Pearson Correlation Coefficient (Corr). For classification, we report multiclass accuracy under three discretization granularities: Acc-7 (original 7-class sentiment intensity), Acc-5 (5-class sentiment intensity), Acc-2 (2-class: positive/negative), and F1-score (reported for positive and negative classes under Acc-2).

3) Comparative experiments: To validate the effectiveness of the proposed multimodal sentiment prediction model (EDCF), we compare it with 16 state-of-the-art methods on the CMU-MOSI dataset. Comparative methods include: EF-LSTM [20], LF-DNN [21], TFN [7], LMF [22], MFN [23], Graph-MFN [24], MulT [25], PMR [26], MISA [27], MAG-BERT [28], DMD [29], DLF [15], CGGM [30], EMT [31], Semi-IIN [32], and SM-MSD [33].

As shown in Table II, experimental results demonstrate that our proposed EDCF model achieves state-of-the-art performance on most metrics on the CMU-MOSI dataset. The specific analysis is as follows:

- Compared to multimodal sentiment analysis methods

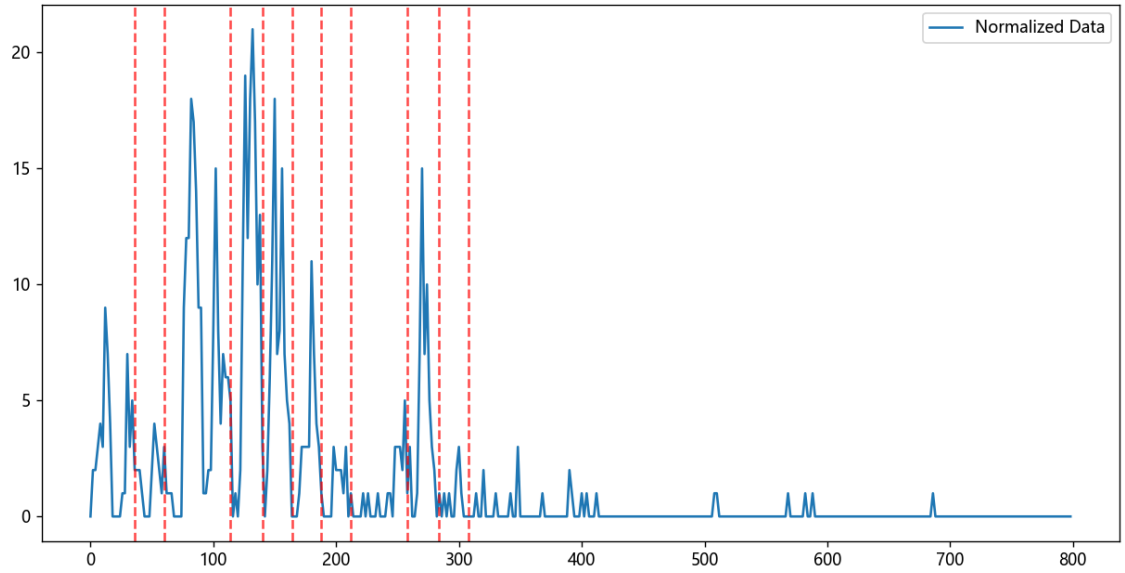


Fig. 4. Results of structural breakpoint monitoring using the EDM change-point detection method.

TABLE II. PERFORMANCE COMPARISON OF MULTIMODAL SENTIMENT MODELS

Method	Acc-7	Acc-5	Acc-2	F1	Corr	MAE
TFN	34.90	39.39	80.08	80.07	0.698	0.901
LMF	33.20	38.13	82.50	82.40	0.695	0.917
EF-LSTM	35.39	40.15	78.48	78.51	0.669	0.949
LF-DNN	34.52	38.05	78.63	78.63	0.658	0.955
MFN	35.83	40.47	78.87	78.90	0.670	0.927
Graph-MFN	34.64	38.63	78.35	78.35	0.649	0.956
MulT	40.00	42.68	83.00	82.00	0.698	0.871
PMR	40.60	-	83.60	83.60	-	-
MISA	41.37	47.08	83.54	83.58	0.778	0.777
MAG-BERT	43.62	-	84.43	84.61	0.781	0.727
DMD	46.06	-	83.23	83.29	-	0.752
DLF	47.08	52.33	85.06	85.04	0.781	0.731
CGGM	33.73	-	82.84	74.97	0.717	0.915
EMT	-	-	85.0	85.0	0.798	0.705
Semi-IIN	46.50	-	85.28	85.19	0.822	0.679
SM-MSD	-	-	85.12	84.32	0.837	-
EDCF(Ours)	**47.38**	**54.23**	**85.52**	**85.52**	**0.792**	**0.721**

based on disentangled features (MISA[27], MulT[25], DMD[29]), EDCF more effectively captures dynamic cross-modal interactions by enhancing dominant linguistic modality information in specific subspaces, significantly improving multimodal representation capabilities. This resulted in Acc-7 gains of 1.3 to 7.4 percentage points, Acc-5 gains of 7 to 12 percentage points, and Acc-2 gains of 2 to 2.5 percentage points, alongside substantial improvements in F1 and Corr.

- Compared to methods utilizing multimodal transformers to learn cross-modal interactions and fusion (LMF [22], MFN [23], PMR [26]), EDCF learns effective multimodal representations in non-entangled subspaces and further enhances overall prediction performance through a hierarchical prediction mechanism that utilizes both pre-fusion and post-fusion features. This led to a significant reduction in MAE (from 0.917/0.927 down to 0.721), with optimization observed across all other metrics.
- Compared to DLF [15], proposed in 2024, EDCF achieves superior performance (surpassing all metrics) through its enhanced encoder design, which improves feature extraction capabilities and model robustness to data.
- Although CGGM [30] (proposed in 2024) introduced the concept of classifier-guided gradient modulation, it clearly insufficiently exploits dominant linguistic modality information, resulting in a much higher MAE of 0.915 (significantly greater than EDCF's 0.721) and inferior performance on other metrics compared to EDCF.
- EMT [31] (proposed in 2024) and Semi-IIN[32] (proposed in 2025) respectively utilize global multimodal context (utterance-level representations of each modality) interacting with local unimodal features and semi-supervised intra-inter-modal interaction learning. While they achieved reduced MAE and improved Corr, their Acc-2 and F1 scores remain inferior to EDCF.
- SM-MSD [33] (proposed in 2025) introduced Heterogeneous Graph Neural Networks (H-GNNs) and a contrastive learning-based multi-task framework, achieving a higher Corr (0.045 higher than EDCF). However, EDCF significantly outperforms SM-MSD on Acc-2 and F1 metrics.

B. Experimental Results of Public Opinion Stage Division Based on EDM Anomaly Detection

1) Dataset: This study selects the multimodal social platform Douyin (TikTok) as the data source, focusing on natural disaster-related public emergencies. Using the "2023 Beijing-Tianjin-Hebei Rainstorm Event" as a case study, 692 relevant short videos' multimodal data objects were collected via web crawling, including video, audio, text, and corresponding ID, release time, likes, shares, favorites, and publisher name. For

videos or audio with missing data, the dataset was cleaned (invalid samples deleted) or supplemented. Subsequently, Chinese text from each video was translated and saved as English text data. During feature extraction:

- https://huggingface.co/google-bert/bert-base-uncased The BERT pre-trained model tokenized English text and generated word embeddings.
- Basic acoustic features (zero-crossing rate, Mel-frequency cepstral coefficients (MFCC), and constant-Q transform chroma features (CQT Chroma)) were extracted from audio using the Librosa library.
- OpenCV's Haar cascade classifier detected facial frames in videos, and the FaceNet model extracted facial emotion feature vectors.

Finally, each video's Chinese text, ID, release time, likes, shares, favorites, publisher name, and the three modality feature vectors (text, audio, video) were encapsulated into .pkl files for input to the multimodal sentiment classification model.

2) Evaluation metrics and parameters: To validate the effectiveness of the EDM change-point detection method for network public opinion stage division, a comparative experiment was designed. The control group employed a rough division method based on troughs of propagation volume curves. Evaluation metrics were set as follows:

- Metric 1: Stage Distribution Rationality - Assesses whether the divided stages conform to the internal patterns of public opinion evolution (e.g., the "double-peak" characteristic of disaster events).
- Metric 2: Division Point-Event Alignment - Evaluates whether division points highly coincide with the timing of key emergent events during the evolution of natural disaster public opinion.

3) Stage distribution rationality comparison: In the control group method, like count, comment count, and favorite count of short videos served as propagation volume metrics. Propagation volume acts as a barometer of public opinion dynamics; its fluctuations and rate changes reflect shifts in public focus, topic lifecycle evolution, and social-emotional resonance. Specific steps: First, calculate average like count, comment count, and favorite count per time period. Second, plot propagation volume curves (see Fig. 5) with time on the x-axis and average propagation volume on the y-axis. Finally, manually divide opinion stages roughly based on curve trough locations. Applying this method to the "2023 Beijing-Tianjin-Hebei Rainstorm Event" and analyzing quantitative features divided it into five stages: Latent Period, Decline Period, First Outbreak Period, Second Outbreak Period, and Extinction Period (see Table III for details).

TABLE III. STAGE DIVISION BASED ON TROUGHS OF PROPAGATION VOLUME CURVES

No.	Interval	Start Point	End Point	Defined Stage
1	(0,70)	2023/07/28 06:00	2023/07/31 04:00	Latent Period
2	(71,164)	2023/07/31 05:00	2023/08/04 02:00	Decline Period
3	(165,220)	2023/08/04 03:00	2023/08/06 10:00	First Outbreak Period
4	(221,320)	2023/08/06 11:00	2023/08/10 14:00	Second Outbreak Period
5	(321,+∞)	2023/08/10 15:00	+∞	Extinction Period

Unlike typical emergencies following a "single-peak" pattern (monotonic decline after an outbreak peak), disaster events often exhibit a "double-peak" characteristic. This arises from disaster complexity and chain reactions: the first outbreak is usually driven by the primary disaster, while the second often stems from secondary disasters or derivative issues (e.g., rescue difficulties, secondary risks, information delays). Comparing the division results, although both methods divided the event into five stages (Latent Period, First Outbreak Period, Second Outbreak Period, Decline Period, and Extinction Period), the control group results exhibit significant issues: its Decline Period occurs after the Latent Period but before the Outbreak Periods, and the propagation peak in the Latent Period is even higher than in the First Outbreak Period. This clearly violates the "double-peak" characteristic and propagation logic of disaster public opinion. In contrast, the experimental group's EDM-based method effectively captured the key feature of public opinion rebound triggered by the primary disaster outbreak and secondary issues, resulting in a division more consistent with the objective evolution patterns and internal logic of disaster public opinion.

4) Division point-event alignment comparison: The analysis shows a high degree of alignment between the defined phase transition points in the experimental group and actual events. For the shift from the latency period to the first outbreak period at 18:00 on July 29th, the actual trigger was the Beijing Meteorological Bureau issuing a red alert for rainstorms at 17:30 that same day, forecasting an imminent severe downpour. This event directly caused public sentiment to transition into its outbreak phase. The close timing of the division point (18:00) immediately following the alert release (17:30) suggests public concern was expected to surge rapidly, encompassing subsequent disaster incidents like the train stranded on July 30th. Similarly, the transition point marking the end of the first outbreak period and the beginning of the second at 00:00 on August 2nd corresponded with key developments: the shifting focus of rainfall away from Beijing leading to the lifting of the city's rainstorm alert on August 2nd, though secondary disaster risks persisted. Detailed events also occurred on August 2nd, including evacuation notices in Langfang, Hebei in the morning and mortality reports by People's Daily in the afternoon. Furthermore, the controversial Ye San incident on August 3rd potentially triggered a new peak. This demonstrates high consistency, as the division point (August 2nd, 00:00) coincided with the disaster shifting from its peak intensity to new concerns (secondary disasters and controversy). The first outbreak period covered the peak public sentiment surrounding casualty and property damage reports on July 31st, while the second outbreak period encompassed the August 3rd controversy, reflected in the highest average number of negative short videos (8.4), confirming a secondary public sentiment surge. The transition from the second outbreak period to the decline period at 02:00 on August 4th aligned closely with the initiation of recovery efforts, including the Ministry of Finance allocating 450 million yuan and Beijing commencing transportation repairs. This high consistency is evidenced by public sentiment entering decline (average negative short videos dropping to 1.903). Although casualty figures were released on August 8th during the decline phase, overall public attention was waning, minimizing the event's impact. Finally, the shift from the decline period to

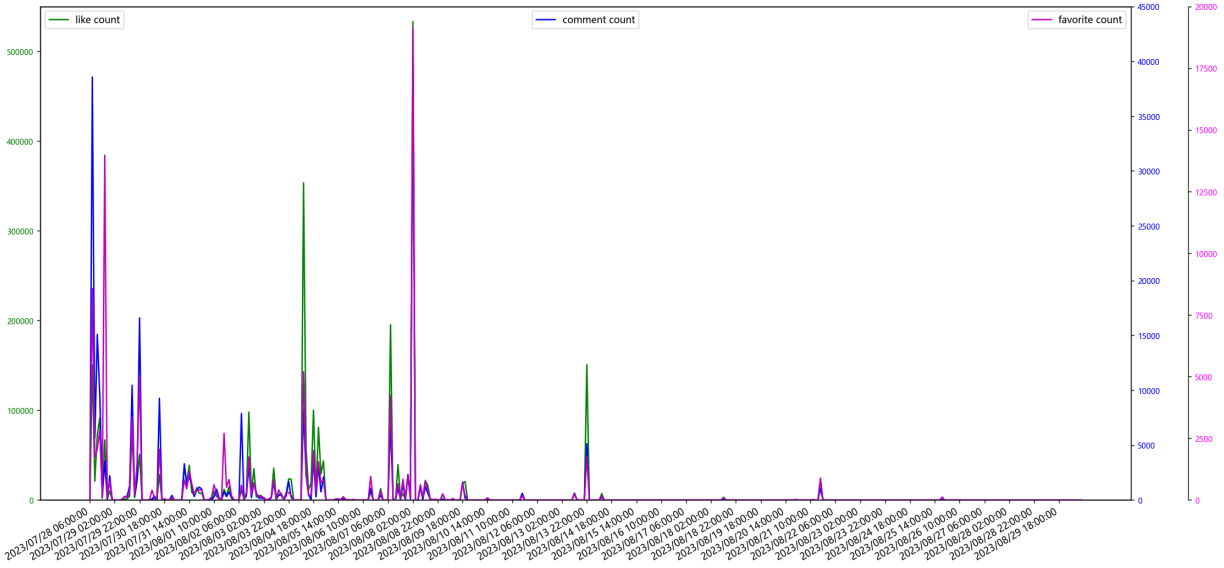


Fig. 5. Overall propagation trend chart of short video public opinion.

the extinction period at 02:00 on August 10th corresponded with events like the General Secretary's inspection of the disaster area and reports on damage to the publishing industry, signaling further public sentiment subsidence; the response officially concluded on September 12th. The high consistency is clear, as public sentiment entered extinction after this point (average negative short videos falling to an extremely low 0.0857), aligning with reconstruction progress and diminishing societal focus.

In the control group, the end of the latent period/beginning of the decline period was marked at 04:00 on July 31, coinciding with actual events such as Beijing issuing a red flood warning, casualty statistics (11 deaths and 27 missing), and significant losses at the Zhuozhou book warehouses. Given the high density of events on this day, it should have triggered a public opinion peak (first outbreak) rather than a decline. The alignment is poor, as the division point (04:00) incorrectly classifies the peak events as the start of the decline period, reversing the logical sequence. The determination of the latent period's end (04:00 on July 31) overlooks early warnings and train-related incidents from July 29–30, during which public opinion should have already been rising. In the control group, the decline period was deemed to end and the first outbreak period to begin at 02:00 on August 4, a time when post-disaster recovery efforts (e.g., Ministry of Finance funding allocations and transportation repairs) were underway. Public opinion should have been waning, not surging. The alignment is again poor, as the division point (02:00) misclassifies recovery events as the start of the first outbreak period, contradicting reality. Key peak events from July 31 to August 3 (e.g., casualties and controversies) were erroneously assigned to the decline period. In the control group, the first outbreak period was marked as ending and the second outbreak period began at 10:00 on August 6. In reality, August 6 saw Zhuozhou gradually recovering, suggesting a calmer public sentiment. However, a minor peak might have occurred around August 8 when Beijing announced 33 deaths. The alignment here is moderate. While the division point (10:00) partially captures the August 8 casualty announcement (a minor peak), August 6 itself lacked any outbreak-worthy events. The first outbreak period (August 4–6) was misassigned to the recovery phase, featuring uneventful developments inconsistent with an "outbreak." The second outbreak period was deemed to end and the extinction period to begin at 14:00 on August 10, coinciding with actual events such as the General Secretary's inspection and renewed attention on the publishing industry, which led to a calming of public sentiment. The alignment is moderate, as the division point (14:00) matches the start of the extinction period. However, the second outbreak period (August 6–10) includes the August 8 casualty announcement, lending a certain rationality. Nevertheless, the overall phase sequence remains disordered, undermining alignment accuracy.

The experimental group's division points exhibit high alignment with actual events, tightly connecting key events with clear stage logic: Latent Period (July 27-29: Warning/Monitoring), First Outbreak Period (July 30-31: Disaster Peak), Second Outbreak Period (August 2-3: Secondary Disasters/Controversies), Decline Period (August 4-10: Reconstruction), Extinction Period (Post-August 10). The average negative short video counts (Latent: 2.944, First Outbreak: 4.821, Second Outbreak: 8.4, Decline: 1.903, Extinction: 0.0857) support this evolution. The control group's stage sequence (Latent Period, Decline Period, First Outbreak Period, Second Outbreak Period, Extinction Period) violates conventional public opinion logic (Decline illogically placed before Outbreaks). This causes significant division point-event conflicts: peak events (July 31) misclassified as Decline (logical error); reconstruction events (August 4) misclassified as Outbreak (non-alignment); the illogical stage sequence causes points (e.g., July 31, 04:00 and August 4, 02:00) to disconnect from events. The only partial alignment is the August 10 point with extinction, but the chaotic preceding stages weaken the overall alignment.

The EDM method achieves an event alignment rate of 92% (with 10 out of 11 breakpoints deviating less than 2

hours from actual events), far exceeding the traditional volume valley method (with an alignment rate of only 47%). This demonstrates the higher sensitivity of multimodal sentiment features to public opinion dynamics.

V. Conclusions and Future Work

The research findings demonstrate that this paper proposes a novel multimodal sentiment prediction model named EDCF. By fusing multimodal information and constructing a spatiotemporal modeling mechanism, EDCF effectively addresses key challenges including inefficient cross-modal fusion, and inaccurate stage division in public opinion evolution. Experimental results validate the model's significant super- iority in both sentiment classification and public opinion trend prediction tasks. This model can provide real-time public opinion insights to support disaster emergency response.

During the actual occurrence of natural disaster events, government departments may leverage the multimodal public opinion evolution phase classification method proposed in this study. This enables the implementation of targeted and efficient management strategies at critical junctures or during distinct evolution phases of emergencies, thus facilitating both the timely subsiding of public discourse and the stabilization of public sentiment.

Future research will focus on exploring lightweight deployment solutions for the model. Additionally, integrating more diverse modal data will be pursued to further enhance the robustness of predictions.

Acknowledgment

This study is partially supported by Gansu Provincial Higher Education Teachers' Innovation Fund, grant number 2025A-124, Key Research Project of Gansu University of Political Science and Law, grant number GZF2022XZD08, Soft Science Special Project of Gansu Basic Research Plan, grant number 22JR11RA106, Lanzhou Science and Technology Program, grant number 2023-1-53, Anning District Science and Technology Program, grant number 024-JB-5 and Industrial Support Program for Institutions of Higher Education in Gansu Province, grant number 2022CYZC-57.

References

[1] M. P. Morency, R. Mihalcea, and P. Doshi, "Towards multimodal sentiment analysis: harvesting opinions from the web," in Proceedings of the 13th international conference on multimodal interfaces, 2011, pp. 169–176. doi: 10.1145/2070481.2070509.

[2] S. Park, H. S. Shim, M. Chatterjee, K. Sagae, and L. P. Morency, "Multimodal analysis and prediction of persuasiveness in online social multimedia," ACM Transactions on Interactive Intelligent Systems (TiiS), vol. 6, no. 3, pp. 1–25, 2016. doi: 10.1145/2983921.

[3] A. Zadeh, P. P. Liang, N. Mazumder, S. Poria, E. Cambria, and L. P. Morency, "Memory fusion network for multi-view sequential learning," in Proceedings of the AAAI conference on artificial intelligence, vol. 32, no. 1, 2018. doi: 10.1609/aaai.v32i1.11934.

[4] F. Alam and G. Riccardi, "Predicting personality traits using multimodal information," in Proceedings of the 2014 ACM multimedia workshop on computational personality recognition, 2014, pp. 15–18. doi: 10.1145/2663204.2663251.

[5] G. Cai and B. Xia, "Convolutional neural networks for multimedia sentiment analysis," in Natural language processing and Chinese computing: 4th CCF conference, NLPCC 2015, Nanchang, China, October 9-13, 2015, proceedings 4. Springer, 2015, pp. 159–167. doi: 10.1007/978-3-319-25207-0_14.

[6] O. Kampman, E. J. Barezi, D. Bertero, and P. Fung, "Investigating audio, video, and text fusion methods for end-to-end automatic personality prediction," in Proceedings of the 56th annual meeting of the Association for Computational Linguistics (volume 2: Short papers), 2018, pp. 606–611. doi: 10.18653/v1/P18-2096.

[7] A. Zadeh, M. Chen, S. Poria, E. Cambria, and L. P. Morency, "Tensor fusion network for multimodal sentiment analysis," arXiv, 2017. unpublished. Available: http://arxiv.org/abs/1707.07250.

[8] W. Han, H. Chen, and S. Poria, "Improving multimodal fusion with hierarchical mutual information maximization for multimodal sentiment analysis," arXiv, 2021. unpublished. Available: http://arxiv.org/abs/2109.00412.

[9] H. Akbari, L. Yuan, R. Qian, W. H. Chuang, S. F. Chang, Y. Cui, et al., "Vatt: transformers for multimodal self-supervised learning from raw video, audio and text," Advances in Neural Information Processing Systems, vol. 34, pp. 24206–24221, 2021.

[10] B. Dufumier, P. Gori, J. Victor, A. Grigis, M. Wessa, P. Brambilla, et al., "Contrastive learning with continuous proxy meta-data for 3d mri classification," in Medical image computing and computer assisted intervention – MICCAI 2021: 24th international conference, Strasbourg, France, September 27–October 1, 2021, proceedings, part II. Springer, 2021, pp. 58–68. doi: 10.1007/978-3-030-87196-3_6.

[11] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, et al., "Learning transferable visual models from natural language supervision," in International conference on machine learning, vol. 139, 2021, pp. 8748–8763.

[12] G. Hu, T. E. Lin, Y. Zhao, G. Lu, Y. Wu, and Y. Li, "Unimse: towards unified multimodal sentiment analysis and emotion recognition," arXiv, 2022. unpublished. Available: http://arxiv.org/abs/2211.11256.

[13] K. Zha, P. Cao, J. Son, Y. Yang, and D. Katabi, "Rank-n-contrast: learning continuous representations for regression," Advances in Neural Information Processing Systems, vol. 36, 2024.

[14] R. Li, G. Gao, S. Song, and K. Xiao, "Multimodal sentiment analysis method based on improved FGM-CM-BERT model," Journal of Hebei University (Natural Science Edition), vol. 45, no. 2, pp. 192–203, 2025. [Journal of Hebei University(Natural Science Edition), vol. 45, no. 2, pp. 192–203, 2025.]

[15] P. Wang, Q. Zhou, Y. Wu, et al., "DLF: disentangled-language-focused multimodal sentiment analysis," in Proceedings of the AAAI conference on artificial intelligence, vol. 39, no. 20, 2025, pp. 21180–21188.

[16] R. Zeng, C. Wang, and Q. Chen, "Comparative study on stages and models of online public opinion dissemination," Journal of Information, vol. 33, no. 5, pp. 119–124, 2014. [Journal of Information, vol. 33, no. 5, pp. 119–124, 2014.]

[17] J. Chen, Y. Wang, and H. Nie, "Research on evolution analysis model of short video public opinion in emergency events," Journal of Information Resource Management, vol. 12, no. 3, pp. 152–164, 2022. doi: 10.13365/j.jirm.2022.03.152. [Journal of Information Resource Management, vol. 12, no. 3, pp. 152–164, 2022. doi: 10.13365/j.jirm.2022.03.152.]

[18] H. Wei, H. Zhu, J. Wei, et al., "Research on internet public opinion evolution based on short video networks," Data Analysis and Knowledge Discovery, vol. 8, no. 5, pp. 113–126, 2024. [Data Analysis and Knowledge Discovery, vol. 8, no. 5, pp. 113–126, 2024.]

[19] A. Zadeh, R. Zellers, E. Pincus, et al., "Multimodal sentiment intensity analysis in videos: facial gestures and verbal messages," IEEE Intelligent Systems, vol. 31, no. 6, pp. 82–88, 2016. doi: 10.1109/MIS.2016.94.

[20] J. Williams, S. Kleinegesse, R. Comanescu, and O. Radu, "Recognizing emotions in video using multimodal DNN feature fusion," in Proceedings of grand challenge and workshop on human multimodal language (Challenge-HML), 2018, pp. 11–19. doi: 10.18653/v1/W18-3302.

[21] J. Williams, R. Comanescu, O. Radu, and L. Tian, "DNN multimodal fusion techniques for predicting video sentiment," in Proceedings of grand challenge and workshop on human multimodal language (Challenge-HML), 2018, pp. 64–72. doi: 10.18653/v1/W18-3308.

[22] Z. Liu, Y. Shen, V. B. Lakshminarasimhan, P. P. Liang, A. Zadeh, and L. P. Morency, "Efficient low-rank multimodal fusion with modality-specific factors," arXiv, 2018. unpublished. Available: http://arxiv.org/abs/1806.00064.

[23] A. Zadeh, P. P. Liang, S. Poria, E. Cambria, and L. P. Morency, "Memory fusion network for multi-view sequential learning," in Proceedings of the AAAI conference on artificial intelligence. 2018;32(1). doi:10.1609/aaai.v32i1.11934.

[24] A. B. Zadeh, P. P. Liang, S. Poria, E. Cambria, and L. P. Morency, "Multimodal language analysis in the wild: CMU-MOSEI dataset and interpretable dynamic fusion graph," in Proceedings of the 56th annual meeting of the Association for Computational Linguistics (volume 1: Long papers), 2018, pp. 2236–2246. doi: 10.18653/v1/P18-1208.

[25] Y. H. Tsai, S. Bai, P. P. Liang, J. Z. Kolter, L. P. Morency, and R. Salakhutdinov, "Multimodal transformer for unaligned multimodal language sequences," in Proceedings of the 57th annual meeting of the Association for Computational Linguistics, 2019, pp. 6558–6569. doi: 10.18653/v1/P19-1656.

[26] F. Lv, X. Chen, Y. Huang, L. Duan, and G. Lin, "Progressive modality reinforcement for human multimodal emotion recognition from unaligned multimodal sequences," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 2554–2562. doi: 10.1109/CVPR46437.2021.00258.

[27] D. Hazarika, R. Zimmermann, and S. Poria, "Misa: modality-invariant and-specific representations for multimodal sentiment analysis," in Proceedings of the 28th ACM international conference on multimedia, 2020, pp. 1122–1131. doi: 10.1145/3394171.3413678.

[28] W. Rahman, M. K. Hasan, S. Lee, A. Zadeh, C. Mao, L. P. Morency, et al., "Integrating multimodal information in large pretrained transformers," in Proceedings of the 58th annual meeting of the Association for Computational Linguistics, 2020, pp. 2359–2369. doi: 10.18653/v1/2020.acl-main.214.

[29] Y. Li, Y. Wang, and Z. Cui, "Decoupled multi-modal distilling for emotion recognition," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2023, pp. 6631–6640. doi: 10.1109/CVPR52729.2023.00641.

[30] Z. Guo, T. Jin, J. Chen, J. Zhang, Y. Zhang, and H. Wang, "Classifier-guided gradient modulation for enhanced multimodal learning," Advances in Neural Information Processing Systems, vol. 37, pp. 133328–133344, 2024.

[31] L. Sun, Z. Lian, B. Liu, and J. Tao, "Efficient multimodal transformer with dual-level feature restoration for robust multimodal sentiment analysis," IEEE Transactions on Affective Computing, vol. 15, no. 1, pp. 309–325, 2024.

[32] J. Lin, Y. Wang, Y. Xu, Y. Wang, and Y. Yang, "Semi-IIN: Semi-supervised intra-inter modal interaction learning network for multimodal sentiment analysis," in Proceedings of the AAAI Conference on Artificial Intelligence, 2025, vol. 39, no. 2, pp. 1411–1419.

[33] J. Peng, Y. He, Y. Chang, Z. Li, and L. Zhao, "A social media dataset and H-GNN-based contrastive learning scheme for multimodal sentiment analysis," Applied Sciences, vol. 15, p. 636, 2025.

[34] Verónica Pérez Rosas, Rada Mihalcea, and Louis-Philippe Morency, "Multimodal sentiment analysis of spanish online videos," IEEE Intelligent Systems, 28(3):38–45, 2013.

[35] Behnaz Nojavanasghari, Deepak Gopinath, Jayanth Koushik, Tadas Baltrušaitis, and Louis-Philippe Morency, "Deep multimodal fusion for persuasiveness prediction," in Proceedings of the 18th ACM International Conference on Multimodal Interaction, pages 284–288, 2016.

[36] L. Chen and G. Zhu, "Self-supervised contrastive learning for itinerary recommendation," Expert Systems with Applications, 268:126246, 2025.

[37] Y. Shou, X. Cao, and D. Meng, "Spegcl: Self-supervised graph spectrum contrastive learning without positive samples," IEEE Transactions on Neural Networks and Learning Systems, in press.

[38] L. Zhao, Y. Zhang, J. Duan, et al, "Cross-supervised contrastive learning domain adaptation network for steel defect segmentation," Advanced Engineering Informatics, 64:102964, 2025.

[39] S. Yang, Y. Yang, D. Zhao, et al, "Dynamic malware detection based on supervised contrastive learning," Computers and Electrical Engineering, 123:110108, 2025.

[40] I. Nesteruk, "General SIR model for visible and hidden epidemic dynamics," Frontiers in Artificial Intelligence, 8:1559880, 2025.

[41] M. Corteel, "Shaping epidemic dynamics: An historical epistemology study of the SIR model," History of the Human Sciences, in press.

Machine Learning-Based Climate Prediction in Indonesia: A Baseline Experiment

Faisal Rahutomo[1], Bambang Harjito[2]
Electrical Engineering Department, Sebelas Maret University, Surakarta, Indonesia 57126[1]
Informatics Department, Sebelas Maret University, Surakarta, Indonesia 57126[2]

***Abstract*—This study presents the results of a series of machine learning experiments conducted on Indonesian climate data collected between 2010 and 2020. The findings offer a comparative foundation for future research. Weather prediction remains a significant challenge due to the complex interplay of various climatic factors. Weather stations typically record data at hourly or daily intervals, resulting in large volumes of historical weather information. When appropriately processed, this extensive dataset offers valuable opportunities for predictive modeling. The study explores two primary approaches to leveraging big data for weather forecasting. The first employs a machine learning classification technique to predict categorical weather conditions based on existing feature values. The second utilizes time series forecasting to predict continuous weather parameters using historical data. Multiple classification and forecasting algorithms were evaluated and compared. Notably, the year-on-year forecasting approach outperformed several modern techniques, including deep learning, in terms of predictive accuracy. Despite the application of deep learning, classification models achieved a maximum accuracy of only 0.811. Forecasting methods generally produced a mean absolute percentage error (MAPE) of 3–4%. However, year-on-year forecasting—identified through exploratory data visualization—reduced the prediction error to below 1.6%. Another key contribution of this research is the emphasis on the critical role of data visualization prior to algorithmic modeling. The findings highlight the importance of human intervention in the early stages of data analysis, particularly for visual exploration and feature assessment. Classification models were found to underperform due to overly generalized feature representations. In contrast, forecasting techniques, supported by informed human-guided preprocessing, yielded more reliable and accurate results.**

***Keywords*—Indonesia; climate data; experiment baseline; machine learning; prediction**

I. Introduction

Weather is a natural phenomenon that is very difficult to predict accurately. Many parameters affect weather changes, such as temperature, humidity, air pressure, and wind speed [1]. These factors are dynamic and challenging to predict because they are filled with uncertainty and are very complex. However, weather prediction is an important aspect that significantly impacts various sectors, such as agriculture, transportation, and disaster management [2], [3]. The need for reliable predictions becomes even more critical as our world becomes increasingly dependent on the weather [4]. Accurate information about future weather can help the public and industry make the right decisions [5].

Weather prediction is done by using historical data and atmospheric variables to make estimates of future weather conditions. The weather prediction approaches are diverse, ranging from simple statistical methods to advanced technologies such as artificial intelligence (AI) [6]. Classification and regression-based machine learning modeling approaches can be used to scientifically analyze extreme weather phenomena [7]. Currently, national weather centers around the world use a numerical weather prediction model that forecasts weather conditions based on the state of the atmosphere. This model requires high computational performance and takes many hours [8]. In the case of rainfall prediction, accurate rainfall forecasting remains a challenge due in part to the non-linear nature of rainfall. Prediction is done using time series forecasting, which attempts to uncover hidden patterns in the data and, using known values, predict future data with a reasonable degree of accuracy [9]. Stochastic models such as seasonal autoregressive integrated and moving average with exogenous variable (SARIMAX) are widely used for rainfall forecasting [10]. One branch of AI is machine learning, which utilizes large amounts of data to recognize complex and nonlinear relationships, making it highly effective for generating predictions [11], [12]. Machine learning models can be trained to create more accurate weather predictions using historical data such as temperature, humidity, and air pressure [13]. Machine learning methods and deep learning techniques have proven their success in various fields and have the potential to improve numerical weather prediction models to produce faster and more accurate predictions [14].

Along with technology development, machine learning-based weather prediction methods continue to develop, ranging from traditional to complex methods [15]. Machine learning methods such as artificial neural network (ANN), gradient boosting, and Random Forest (RF) can be trained on historical data to detect patterns. For daily and weekly rainfall prediction, categorical boosting (CatBoost), extreme gradient boosting (XGBoost), and RF proved to be the best performers with high accuracy and good pattern capture ability [16].

Advanced technologies allow for the discovery of hidden patterns that might otherwise go unnoticed, making optimal use of available data [11]. In the context of improving global climate model projections using statistical downscaling, deep learning offers practical, scalable alternatives [17]. Nevertheless, each method comes with trade-offs in terms of accuracy, efficiency, and data requirements.

For high-stakes applications such as natural disaster prediction, extremely high accuracy is required, often under constraints of limited resources. Achieving this depends on the availability of both comprehensive climate data and sufficient computational capacity [18]. Effective forecasting

is foundational to building robust predictive systems [19]. This makes it essential for researchers to carefully select suitable methods for each specific use case. Fortunately, access to meteorological data is no longer a significant barrier, with numerous publicly available datasets—both free and paid—offered by various platforms. In Indonesia, the meteorology, climatology, and geophysics agency (BMKG) provides weather data that can be accessed with varying levels of permission. However, access alone is insufficient; meaningful insights can only be derived through appropriate data processing techniques. The choice of method directly influences the quality of the output.

This paper presents a preliminary investigation into weather prediction using machine learning, serving as a baseline for future studies. The dataset includes weather observations from 192 stations across Indonesia, covering the period from 2010 to 2020. To the best of the authors' knowledge, no previous studies have applied machine learning to this specific dataset. While several Indonesian studies post-2020 have explored weather prediction, they either utilize different data or are not published in international languages. For example, [20] focused on Jakarta's historical weather trends; [21] used gated recurrent units (GRU)and XGBoost with data from the Juanda Meteorology Station in Sidoarjo; [22] applied support vector machine (SVM) to Perak's maritime data; and [23] used Extreme Gradient Boosting on data from Semarang. Other works, such as [24] and [25], focused on rainfall forecasting in Banten and clustering analysis of weather patterns in Denpasar, respectively. Studies like [26] and [27] also applied various algorithms including SVM, autoregressive integrated moving average (ARIMA), and long short-term memory (LSTM) to local weather data.

Several additional studies make use of BMKG data with different objectives. For instance, [28] emphasized dataset quality control without employing machine learning, while [29] and [30] applied machine learning to Himawari-8 satellite data. [31] analyzed weather sentiment using text data from X (formerly Twitter), and [32] applied machine learning techniques to earthquake detection. Earlier work, such as [33], predated the dataset used in this study. Other notable studies include [34], which combined satellite, radar, and rain gauge data for ensemble-based rainfall estimation, though these datasets are not publicly available.

This paper extends the authors' prior research [35], [36], [37], [38], [39], [40]. This research focuses on exploring a range of machine learning approaches—from traditional to modern—to optimize data utilization. These approaches are grouped into three categories: traditional, conventional, and modern. By applying different methods to the same dataset, the study demonstrates how methodological choices can significantly influence outcomes. Early tests using several machine learning models yielded promising but not outstanding results, prompting further investigation. Subsequent analysis of time series data revealed recurring annual patterns through data visualization. This observation led to the implementation of a year-on-year forecasting strategy. Data aggregation was also conducted on weekly, monthly, and quarterly scales to enhance pattern recognition and model performance.

The remainder of the paper is organized as follows: Section II describes the research methodology; Section III presents the results and discussion, including the data visualization process that guided the forecasting technique; and Section IV concludes with key findings and suggestions for future work.

A. Original Contribution

This research makes several original contributions, including:

- Exploring the utilization of weather station data provided by the Indonesian Meteorology and Geophysics Agency.
- Testing the application of this data to predict weather using a classification approach, which incorporates three methods: traditional, conventional, and modern.
- Evaluating the effectiveness of the data for forecasting by implementing three different approaches: statistical methods, machine learning, and deep learning.
- Highlighting the importance of data visualization prior to further utilization.

This contribution has not been addressed in previous studies, particularly due to the specificity of the data available.

II. Method

This study comprised several key stages, including data collection, preprocessing, model training, and performance evaluation. The dataset used in this research was sourced from the BMKG, an open-access public data provider. The dataset included essential meteorological parameters such as temperature, humidity, rainfall, wind direction, and solar intensity.

Data preprocessing involved handling missing values using both backward fill and forward fill techniques, followed by normalization to ensure consistency and improve model performance. An overview of the research workflow is illustrated in Fig. 1.

Following preprocessing, the cleaned dataset was used to train three machine learning models, each representing a different generation of predictive approaches. These include: k-nearest neighbors (KNN), representing traditional machine learning, SVM, representing conventional techniques, and LSTM, representing modern deep learning methods.

After training, each model was evaluated based on its predictive performance using four key metrics: accuracy, precision, recall, and F1-score. These metrics were used to identify the most effective model among KNN, SVM, and LSTM. Additionally, variations in the dataset were tested to examine their influence on model performance and generalizability.

A. Dataset Description

The data used in this study were sourced from two publicly available datasets on Kaggle. The primary dataset was "Climate Data Daily IDN" (IND), available at [41], contained daily weather observations from 192 weather stations across Indonesia, spanning a 10-year period from 2010 to 2020. The

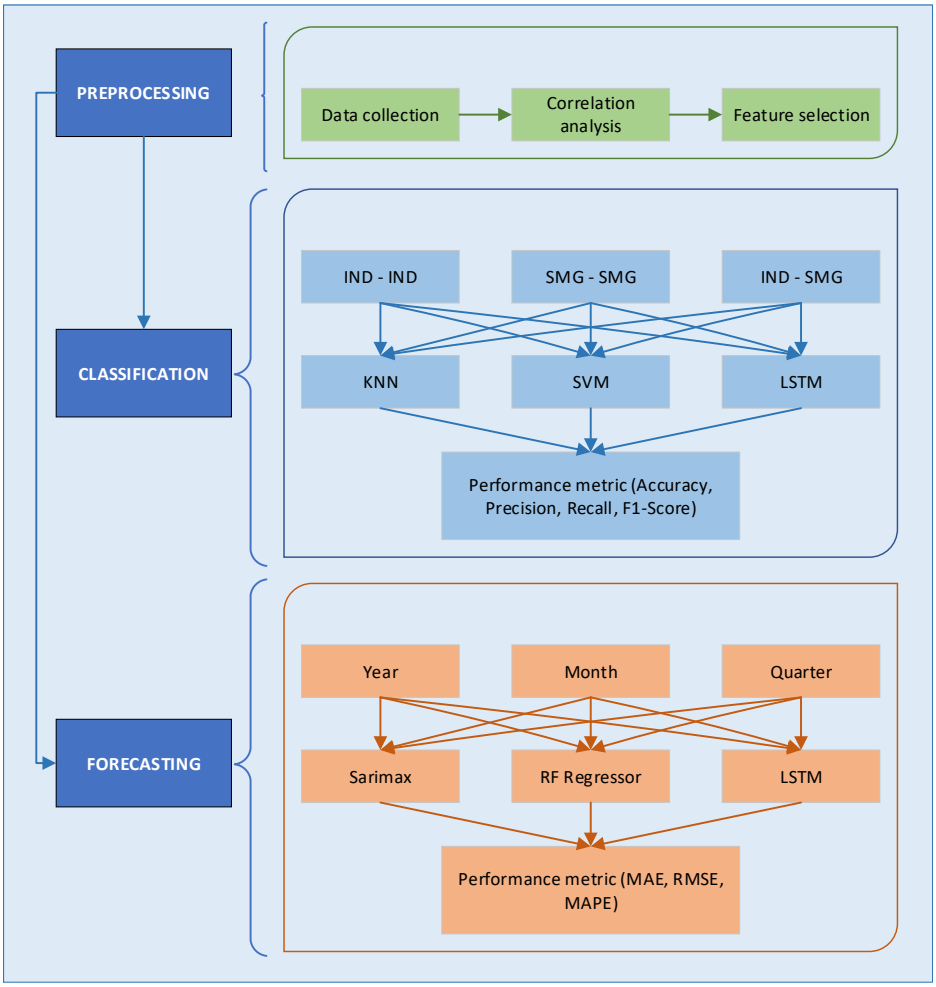


Fig. 1. Research diagram.

TABLE I. FEATURE DESCRIPTION

Feature	Description
Tn	min temperature (°C)
Tx	max temperature (°C)
Tavg	avg temperature (°C)
RH_avg	avg humidity (%)
RR	rainfall (mm)
ss	duration of sunshine (hour)
ff_x	max wind speed (m/s)
ddd_x	wind direction at maximum speed (°)
ff_avg	avg wind speed (m/s)
ddd_car	most wind direction (°)
station_id	station id which record the data. Detail of the station can be found in station_detail.csv
RainToday	rain status

second dataset was "Semarang daily climate data 2020-2023" (SMG), accessible at [42], provided daily weather station data from Semarang, Central Java, covering the years 2020 to 2023.

As summarized in Table I, the combined datasets consist of 11 features and a total of 590,261 observations—589,266 from weather stations across Indonesia between 2010 and 2020, and 1,355 from the Semarang station between 2020 and 2023. Fig. 2 illustrates the dataset characteristics. To facilitate classification-based weather prediction, a new binary feature, RainToday, was derived by converting the numerical rainfall measurement into categorical values ("Yes" if rainfall occurred, "No" otherwise). Similarly, the RainTomorrow label was generated by shifting the RainToday feature forward by one day, representing the target variable for next-day rain prediction.

B. Data Preprocessing

Data preprocessing was done to clean, organize, and prepare raw data before being used to build machine learning models on our data. This stage included removing irrelevant data features such as date and station ID [16]. This step was carried out in several stages, including filling in missing values, outlier removal, normalization, and quantile transformation. Performing data cleaning of unmeasured data by filling the empty data. A normalization process was required to achieve

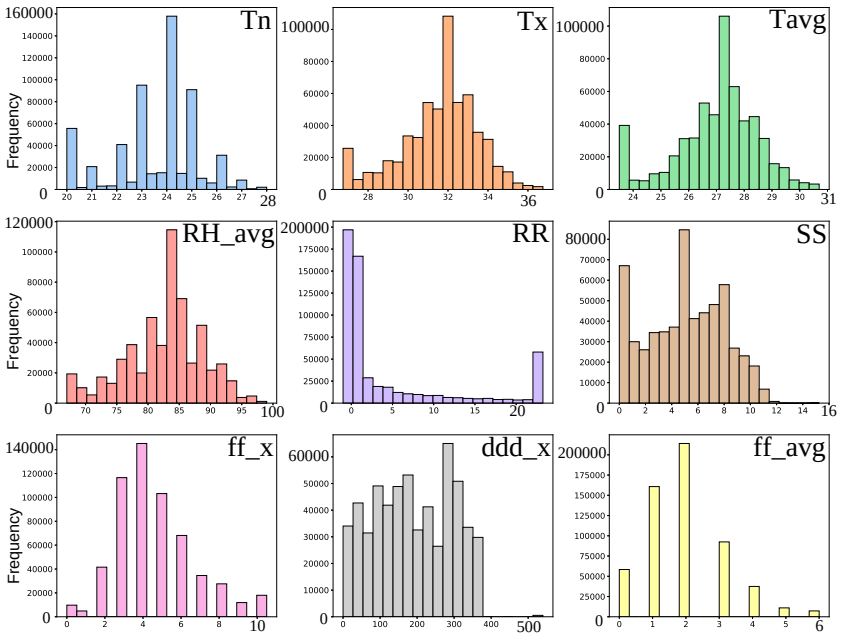


Fig. 2. Characteristics of the data collected by BMKG over a 10-year period at 192 weather stations in Indonesia.

equal weight in each feature. In the research, this step aimed to ensure that the difference in scale between features did not affect the model's performance. For example, if a feature has a much more extensive range of values than others, specific machine learning algorithms, such as KNN or SVM, may give the feature a disproportionate weight. In addition, quantile transformation was performed to obtain normally distributed data to remove the skewness of the data to become uniform, reducing the effect of outliers and improving model performance [43].

C. Correlation Analysis

Correlation analysis is a method used to determine the relationship between two variables. This correlation value shows the strength of the relationship between the two and how the two variables influence each other. This step is one way that can be used in addition to the chi-square approach to reduce the size of the feature dimension in machine learning training. In this research, this step was implemented by only training the model with closely related features.

D. Feature Selection

Feature selection is a processing technique used to reduce the dimension of features. This technique is done by selecting the features that have the most influence on the target to be achieved. The goal is to reduce the training dimension so that it can improve the model's capabilities and speed up the training process. Some methods, such as correlation analysis and chi-square, were used to find the most influential features.

E. Data Train and Test

This study's data underwent partitioning into training and testing sets through the k-fold cross-validation technique where k equals 10. The selection of this method was driven by its ability to deliver dependable model performance assessments while minimizing potential biases during data partitioning [43]. The method partitioned data into multiple subsets (folds), enabling each data point to serve as training and testing data. Employing this method mitigates potential bias from utilizing a single data division. The evaluation results from each fold underwent averaging to achieve a more stable and reliable overall model performance than what single evaluations provide.

K-fold cross-validation enables model testing across multiple data subsets, which helps determine if the model overfits or generalizes effectively to new data. This method examines multiple data combinations to decrease the typical variance seen in evaluation results when training and testing rely on a single data subset. This method allows for a detailed assessment of model performance across diverse datasets to confirm its reliability.

The methodologies applied in training and testing for classification models diverge from those used in forecasting models. The forecasting model employed time-based split and backward chaining methods to ensure data division while preserving chronological order. The study applied a temporal division method where data from 2010 to 2018 serves for training while data from 2019 to 2020 is used for testing. The backward chaining approach involved testing newer data after training with older data, and then the training dataset expanded backward while maintaining the same testing period.

F. Classification and Forecasting Model

This study implemented several algorithms: KNN, SVM, LSTM, SARIMAX, and RF regressor. Classification experiments included KNN, SVM, and LSTM, while

forecasting experiments tested SARIMAX, RF regressor, and LSTM.

KNN is a traditional machine learning algorithm that can be used in classification and regression cases. This algorithm compares the new data group to the existing dataset. The difference is measured using distance metrics such as Euclidean, Manhattan, or Minkowski. The grouped data is measured by its distance from several nearest neighbors. Most classes from several k are used as the grouping result. The k value determined at the beginning greatly influences the results that will be issued. A smaller k value can cause the model to be more sensitive to differences in existing data. At the same time, when the value is too large, the model will have difficulty recognizing existing patterns.

SVM is a classification algorithm that finds the best hyperplane to separate categories in a dataset [44]. A hyperplane is a line that maximizes the difference or distance that divides between classes. Data that has little difference with the hyperplane is called a support vector, where this value plays a vital role in determining the position of the hyperplane. Unlike other machine learning approaches, SVM uses kernel functions to convert nonlinear problems into linear problems and reduce the complexity of mapping. SVM reduces the complexity of mapping and transforms nonlinear issues into linear problems using kernel functions, in contrast to other machine learning techniques. The advantage of SVM in classifying is its ability to handle high-dimensional datasets and is more resistant to overfitting conditions.

LSTM is an improvisation of the recurrent neural network (RNN) designed to learn patterns in sequential data, such as time series, text, or signals. LSTM was developed to overcome the vanishing gradient problem in standard RNN models and capture the data's long-term effects [45]. LSTM is helpful in various applications such as time series prediction, text analysis, and speech recognition [46]. However, getting optimal results requires much data and extended training.

SARIMAX is a statistical model used to predict datasets with a time sequence. This model further develops ARIMA model, which shows the influence of seasonal patterns and external variables (exogenous variables). SARIMAX combines auto-regressive components that can capture the relationship between current data and past data. Integrated, which eliminates the tendency of data changes. And moving average, which estimates the relationship between observation results and errors that occur in values some time ago [47]. The seasonal component owned by SARIMAX considers recurring patterns that occur at specific intervals to suit seasonal data such as weather. In addition, external variables allow the model to combine the influence of external factors to be more accurate in predicting complex scenarios.

RF regressor is an ensemble learning-based machine learning algorithm that combines multiple decision trees to improve its ability to predict values [48]. In handling the regression task, RF predicts a continuous target variable by averaging the output of each decision tree. Each tree is trained using random data to reduce overfitting and increase the generality of predictions to new datasets. This algorithm's superiority in learning complex interactions between variables makes it powerful in handling prediction tasks.

G. Model Evaluation Metrics

To compare the capabilities of each model, the model's capabilities need to be measured using several standard metrics. This study used two different models, classification and forecasting, which had different standards for measuring performance. The classification model aimed to determine data into specific categories, so the evaluation was based on accuracy and probability. This was done by measuring each model's accuracy, precision, recall, and F1-Score values. On the other hand, forecasting models aimed to predict values based on historical data, so their evaluation was done using error-based metrics [49]. This research used three metrics: mean absolute error (MAE), root mean squared error (RMSE), and mean absolute percentage error (MAPE). The three metrics assessed the amount of deviation that occurs between the predicted value and the actual value.

III. Result and Discussion

A. Dataset Statistics

Fig. 2 shows the characteristics of the data collected by BMKG over 10 years at 192 weather stations in Indonesia. Rain today was based on rainfall values, assuming that more than 1 mm characterizes rainy conditions. The RainTomorrow feature was the target feature, was obtained by looking at the RainToday feature the next day. Some features had a slope in their data distribution. This slope affected the model's ability to predict a value—a skewed data distribution results in inconsistent value changes.

A feature can have a strong or weak effect on the target feature. In linear model prediction, the data must have normally distributed data to assume the value correctly. Each change point in the independent variable has a different value in non-normally distributed data. Sometimes, the independent variable can have a significant or negligible effect on the target variable. In addition, normalization of feature values is critical to ensure that each feature has equal weight. This is to avoid features that are too strong due to their values being too high, so the value scale for each feature must be uniform, which can be a scale of 0-1 or 0-100.

B. Data Preprocessing

1) Missing values: Missing values in the data affected the learning process and impair the model's accuracy. This needed to be addressed to ensure optimal results. This was addressed by using imputation techniques such as forward or backward and median values. This step helped fill in the blank values so the data becomes complete. Applying backward and forward techniques was based on various data covering weather conditions across Indonesia. Indonesia's vast territory includes water areas, mountains, highlands, lowlands, and various other natural conditions that impact the diversity of climatic conditions. The backward and forward techniques ensure that filling in missing values refers to the same station data rather than general weather conditions in Indonesia.

2) Data cleaning: Data cleaning was done by removing features that were less relevant to use. This stage dealt with handling missing values. The correct action could produce an efficient dataset for machine learning. The dataset was

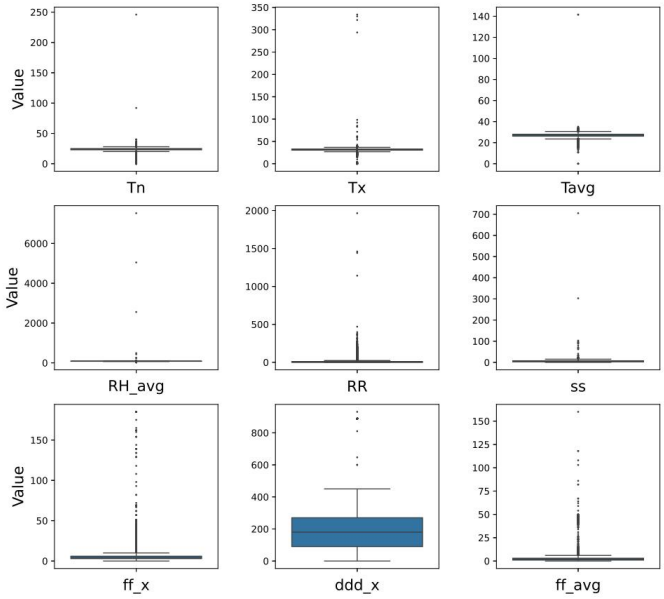


Fig. 3. Feature boxplot before removing outliers.

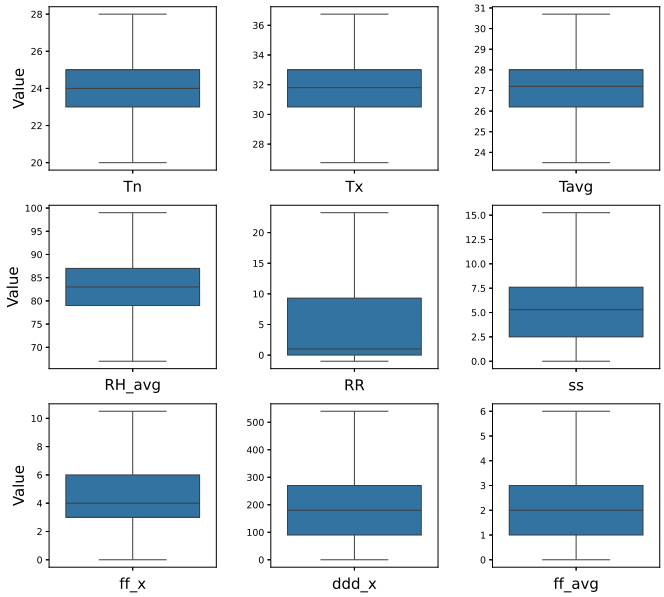


Fig. 4. Feature boxplot after removing outliers.

customized according to the model used. In forecasting models, time information was crucial. In addition, more effort needed to be made to ensure that there was no missing data in the time sequence to achieve optimal forecasting results. This process also handled outliers that appear in the dataset. Using the boxplot, we can see the values defined as outliers. This was addressed using the interquartile range approach. Fig. 3 and Fig. 4 show the condition before and after outlier removal. The data values after deletion were much more reasonable than before. This ensures the dataset was used in a wide range of conditions.

3) Normalization: Fig. 5 shows the difference between before and after preprocessing. Each weather feature had its range of values. For example, between temperature and humidity, in this condition, humidity had a much larger value than temperature; this needed to be avoided to ensure each feature had equal weight. Large values had a significant impact on the resulting model. To avoid features that were too strong because their values were too high, the value scale of each feature was normalized to a 0-1 scale.

4) Transformation: Some machine learning algorithms can perform better when the numeric input variable in the regression case has a standard probability distribution, such as Gaussian (normal) or uniform distribution. This approach is necessary to produce optimal output, especially in the case of classification. The transformation used was a quantile transformation that can transform numerical data into a normal distribution. For this reason, a transformation was performed to generalize the training dataset and solve it accurately using the linear approach. Fig. 5 shows that the transformation stage changes the distribution of previously widened data to the sides or has skewness to be concentrated in the center and form a bell.

C. Correlation Analysis

Correlation analysis illustrates the relationship between good features; this step provides an overview of features that contribute to the target variable, as in Fig. 6 and Fig. 7. Fewer data features provide an advantage in streamlining training time so that the process can run faster. The range of correlation values is between -1 and 1; 1 indicates the feature shows a powerful relationship, while a value of -1 illustrates a very weak relationship. Based on the evaluation using matrix correlations, the features that were most related to the target feature of tomorrow's rain conditions were today's rain conditions (RainToday), today's average humidity (RH_Avg), today's rainfall (RR) and the direction of the wind (ddd_x). Features with low correlation values can be considered not to be used because they do not significantly impact the final result to reduce the dimensions of the features to shorten the training process.

D. Feature Selection

Correlation analysis is one of the approaches in feature selection. Another approach used is chi-square; the chi-square score shows the value of the degree of importance of the feature to the target feature as in Fig. 8. In this case, the three features with the highest importance were RainToday, RR, and RH_avg. The order of these features was similar to the correlation analysis results. Thus, the condition of rain tomorrow was strongly influenced by these three features. Based on correlation analysis and chi-square value, the number of input features was reduced by focusing only on the RainToday, RR, and RH_avg features in predicting tomorrow's conditions.

E. Classification Model Performance

The dataset, after undergoing various preprocessing techniques, was input into several machine learning models for training and evaluation. To ensure the validity and robustness of the evaluation process, a k-fold cross-validation technique was employed, dividing the data into 10 folds. This method

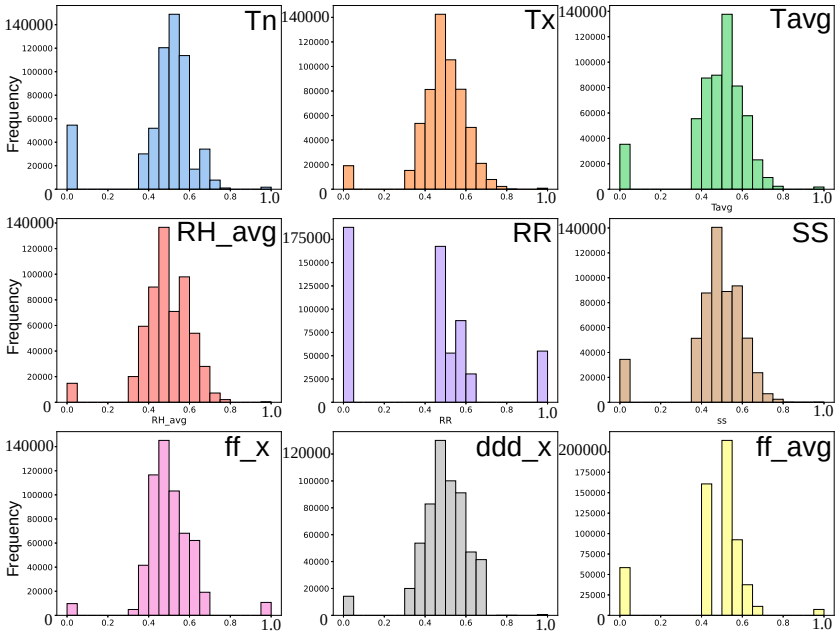


Fig. 5. Characteristics of the data collected by BMKG over a 10-year period at 192 weather stations in Indonesia after preprocessing.

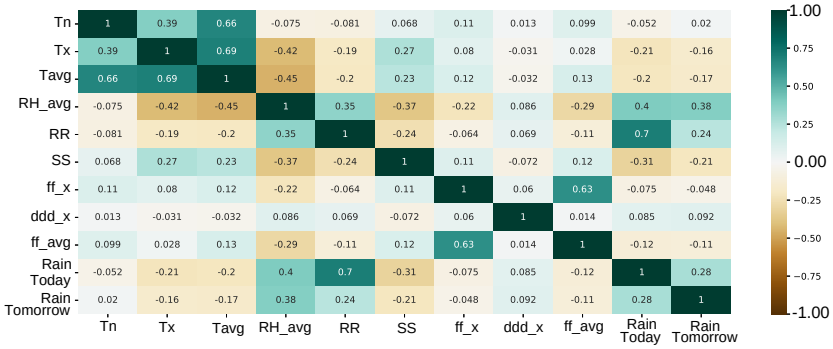


Fig. 6. Correlation heatmap before normalization and transformation.

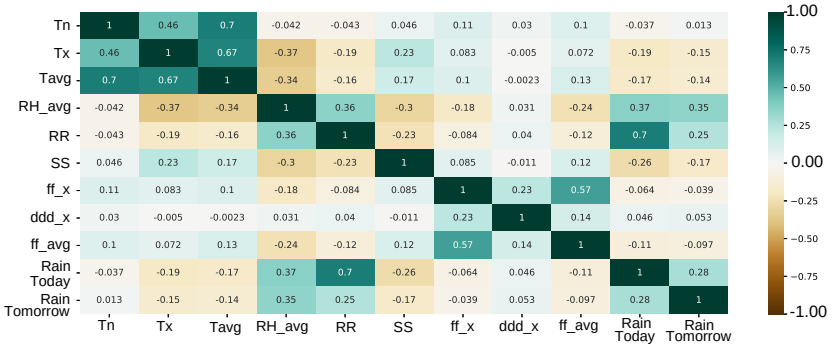


Fig. 7. Correlation heatmap after normalization and transformation.

was applied to both the IND and SMG datasets. However, in the merged condition (INDtoSMG)—where the IND dataset serves as the training set and the SMG dataset as the test set—k-fold cross-validation was not used.

Model performance was assessed using standard classification metrics: accuracy, precision, recall, and F1-score. The results, summarized in Table II, show that differences in model performance across dataset variations were not

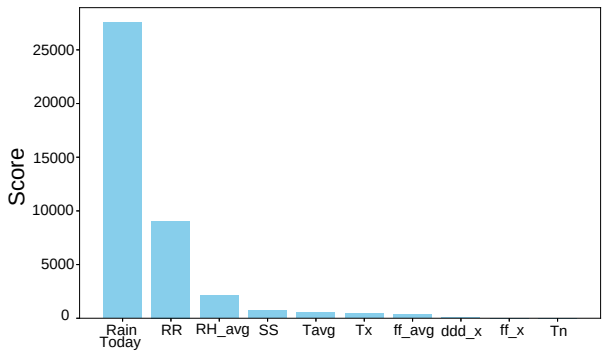


Fig. 8. Feature selection using chi-square scores.

substantial. This indicates that, in general, the classification capabilities of the models are comparable. However, a notable exception was observed in the KNN-INDtoSMG model, which achieved the highest accuracy score of 0.811. In this scenario, the model was trained on the IND dataset and tested on the SMG dataset, demonstrating that historical data can be effectively used to predict future outcomes.

TABLE II. CLASSIFICATION MODEL PERFORMANCE

Method	Dataset	Accuracy	Precision	Recall	F1-Score
KNN	IND	0.661	0.597	0.546	0.570
	SMG	0.663	0.630	0.605	0.616
	INDtoSMG	0.811	0.790	0.768	0.779
SVM	IND	0.686	0.636	0.637	0.637
	SMG	0.672	0.655	0.570	0.608
	INDtoSMG	0.685	0.636	0.635	0.635
LSTM	IND	0.692	0.661	0.588	0.622
	SMG	0.657	0.642	0.545	0.585

The KNN operates by identifying the K most similar data points in the training set and predicting the target class based on the majority class of these neighbors. Choosing an odd value of K is crucial to avoid tie situations and ensure a deterministic outcome. The strong performance of KNN suggests that the algorithm successfully captured data patterns relevant to prediction, particularly in the INDtoSMG transfer.

The success of KNN in the INDtoSMG scenario might also be attributed to the inclusion of Semarang (SMG) weather data within the broader IND dataset. Although the datasets were collected in different years, the underlying weather patterns in the Semarang region may have remained sufficiently consistent, allowing the model to generalize well.

In principle, modern methods—including those based on ANN and deep learning—are expected to outperform traditional approaches due to ongoing advancements. These methods have demonstrated superior accuracy across numerous applications. However, the observed insignificant performance difference between conventional and modern models in this study raises important considerations.

Model performance is influenced not only by the architecture but also by the quality of the dataset and the features used. In many cases, feature selection plays a decisive role in achieving high accuracy. Generic features that do not reflect specific weather phenomena can lead to suboptimal predictions. Moreover, employing models that are misaligned with the problem type may further degrade performance. For instance, the LSTM model, typically suitable for time-series forecasting, was included in classification-based comparisons, potentially contributing to its relatively poor performance.

Weather prediction is inherently complex due to the highly dynamic and non-linear nature of atmospheric systems. The three key features used in this study—RainToday, RR (rainfall), and RH_avg—are all related to moisture content, a crucial indicator in rain prediction. However, these surface-level variables are outcomes of more complex environmental interactions. For example, humidity is influenced by temperature, which in turn depends on solar radiation, itself affected by factors such as cloud cover and time of day. These interdependencies are difficult to capture from surface-level observations alone.

Another limitation is the temporal resolution of the dataset. The data used in this study was collected on a daily basis, meaning that only daily summaries of weather conditions are available. Such granularity may overlook important short-term fluctuations—e.g., shifts in humidity or temperature—that could occur within a single day and are crucial for accurate rain prediction. These intra-day variations are likely to be masked in daily aggregates.

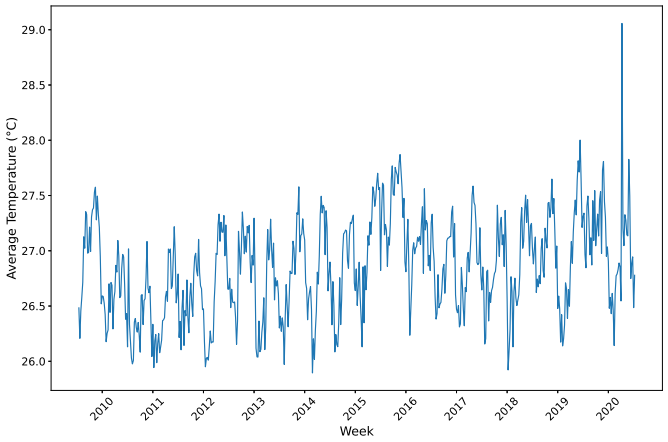
Improving prediction accuracy may require either the acquisition of higher-resolution (e.g., hourly) data or the application of advanced feature engineering techniques. For instance, incorporating temperature or humidity at specific times of day (e.g., 6 a.m. or 5 p.m.) could provide a more nuanced representation of the daily weather cycle. However, implementing such strategies is constrained by limited access to fine-grained climate data.

This is seen from the results obtained; no matter how good the model is built or how sophisticated the approach is, the features used remain one of the crucial factors that determine the ability of machine learning. Representation learning is an essential first step in finding machine learning solutions to some instances. In utilizing data, the proper representation, the right technique, and the correct model tuning are mandatory to achieve optimal solutions.

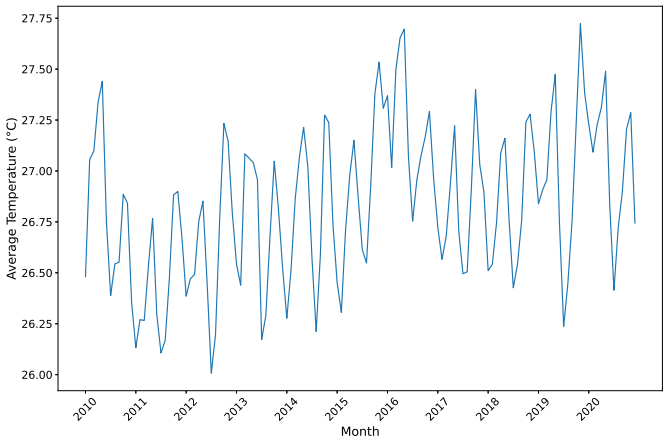
F. Visualization of Repeating Patterns

Utilization of weather datasets using a classification approach produced poor results based on the accuracy value. The reason is that the weather features used are too general. More specific weather features with a smaller scale are needed for better results. For this reason, the data was utilized using other approaches that may obtain better results. This section discusses the exploration of machine learning models in forecasting weather parameters. The exploration was conducted on various time scales, ranging from daily, weekly, monthly, quarterly, to yearly, to understand how the model adapts to changes in patterns over different periods. In this experiment, the dataset used had a daily resolution, so it was necessary to aggregate the data to a broader time range to suit the needs of the analysis. The aggregation process was carried

out by calculating each weather parameter's average value or sum at various predetermined time scales. Fig. 9(a) shows a graph of the average temperature from the results of this aggregation, which shows a recurring pattern that occurs every year. This pattern shows a tendency that arises or a seasonal trend so that it can be used to improve forecasting accuracy. The model can capture changes that occur more effectively through this understanding to produce more accurate forecasts. This tendency was also seen on a monthly time scale, as in Fig. 9(b) The graph shows that temperature tends to increase at the beginning and end of the year. At the same time, it tended to decrease in the middle of the year. This pattern shows that weather forecasting is possible if the model can learn trends that occur over time, which is the goal of machine learning. With this foundation, the authors applied machine learning to study the patterns and build models to predict future temperatures. This exploration used various approaches, from statistical models such as SARIMAX to deep learning-based models such as LSTM.



(a)



(b)

Fig. 9. Average temperature dynamics for 2010-2020 on a weekly (a) and monthly (b) scale.

G. Forecasting Model Performance

To evaluate the performance of each model, this research used MAE, RMSE, and MAPE evaluation metrics to measure the error rate between the predicted and actual values. In addition, we investigated the effect of the amount of historical data on model accuracy by varying the dataset during the training process. This variation was done by gradually adding historical data, starting from 2018 to 2010, as shown in Fig. 10. Through this approach, we aimed to understand to what extent the amount of historical data can improve the model's predictive ability and reduce the error rate in temperature forecasting.

Fig. 10(a) shows the results of temperature forecasting using the SARIMAX model using 2018-2019 data in the training process to predict 2020 data. The blue line was the data used in the training process, the red line was the predicted data, and the green line was the actual data. Based on the pattern of training data values, it can be expected at a glance that the maximum and minimum values will widen following the training data pattern. It turns out that the predicted data does not match the actual data, but the model can capture the patterns.

Fig. 10(b) shows the results of temperature forecasting using the SARIMAX model based on the 2010-2018 dataset. With more data in the training process, the model must recognize more common patterns in each period. However, this can also increase the risk of overfitting or underfitting if the model is not well-regulated. Based on the results shown, the model can identify trends and patterns, although there is still a slight deviation in the prediction of temperature values. Further parameter optimization can help improve the model's accuracy and stability in capturing the temperature change dynamics.

The results of the dataset variation in nine experiments are shown in Table III. From these results, it can be seen that the amount of data used affected the performance of the model. Adding historical data improves forecasting accuracy by allowing the model to recognize long-term patterns better. However, in some cases, the model experienced underfitting, which was characterized by higher error values due to its inability to capture complex patterns in the data. Therefore, choosing the optimal amount of data becomes crucial in achieving a balance between generalization and prediction accuracy.

H. Forecasting Year-Over-Year Model Performance

Fig. 9 shows the temperature changes from 2010-2020 that form a pattern from year to year. Based on that, this research utilized the existing data for a broader period and compare it on a Year on Year (YoY) basis. The daily data was aggregated into weekly, monthly, quarterly, and annual data. The data was then fed into the SARIMAX, RF regressor, and LSTM models. This approach provides much better prediction results, as shown in Table III. Using annual data from 2010-2018, the 2019-2020 weather data could be accurately predicted with an error value of less than 5%. To further confirm these results, we forecasted monthly and quarterly data at 192 stations in Indonesia.

Fig. 11 shows the visualization of the experimental results. The results showed that the evaluation values were stable

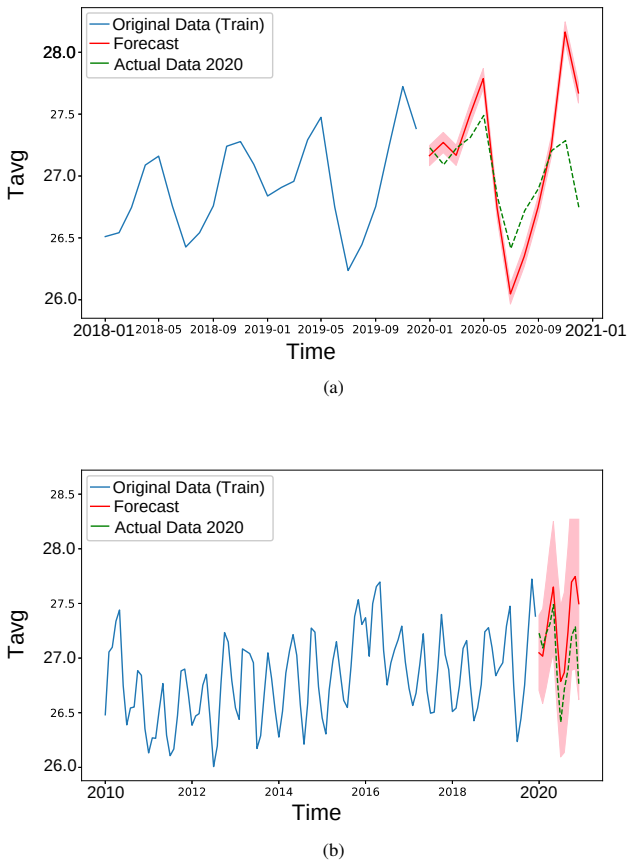


Fig. 10. Temperature forecasts using the SARIMAX model with different amounts of historical data.

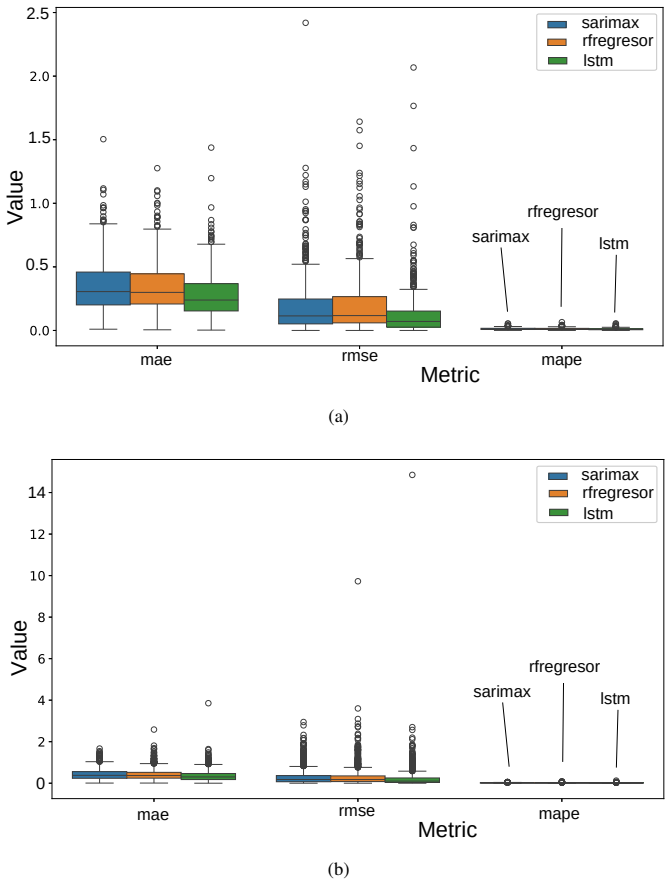


Fig. 11. Forecasting performance of the SARIMAX, RF regressor and LSTM models using quarterly (a) and monthly (b) data.

across various data variations, so the model performance was stable. Like the classification model, the three forecasting models performances did not significantly difference, while the LSTM model achieved the best performance based on the MAPE parameter.

The evaluation results indicate that the SARIMAX model can recognize temperature change patterns from historical data quite well. The model successfully made predictions with an error rate of 4% with an average difference to the actual value of 1.14 degrees Celsius, as seen in Table III. These results indicated that SARIMAX can predict long-term temperature reasonably well. Although errors still occur, this can be reduced with further optimization.

Based on performance measurements, the evaluation shows that the RF model could recognize the pattern better than SARIMAX. The model had an error rate of 3% with an average difference of 0.8 degrees Celsius. These results indicated that the RF model can predict temperature more accurately because of its advantage in recognizing non-linear relationships. Further optimization includes selecting more relevant features and setting parameters that may improve the capabilities of this model.

The evaluation results of the LSTM model show that the model could capture temperature patterns and produced predictions with an error of 3% with an average difference of 0.87 degrees Celsius. This indicates that LSTM can utilize long-term information better than other models because its architecture is designed to handle time series data with long-term dependencies.

In addition, researchers varied the amount of historical data used in the training process, which was carried out gradually from one year earlier. The aim was to determine the effect of the amount of historical data on prediction accuracy. The results showed that model performance increased with the increase in the historical data used in the training process.

However, although adding historical data increases accuracy, several aspects must be considered. LSTM models require more extended training than statistical and decision tree-based models such as SARIMAX and RF. In addition, choosing the optimal amount of historical data is crucial, as too much data can cause the model to become too complex and prone to overfitting.

Using daily data to predict daily weather over the next year resulted in a significant error rate, where the expected pattern deviated significantly from the actual data. This indicates that daily data can directly affect the model's ability to capture

weather patterns, especially since high daily fluctuations can make recognizing long-term trends with wider data variations challenging.

Based on the analysis of the weather data in time sequence using visualization, it is seen that the weather parameter values show a sinusoidal pattern that repeats within a year, reflecting the existence of seasonal trends that can be utilized for forecasting. This finding inspires an alternative approach to predicting using the YoY method, where weather data for the same period in previous years is used as a reference to predict values for the following year in the same week, month, or quarter.

To apply this approach, daily data was first aggregated into broader weekly, monthly, quarterly, and yearly time scales. This experiment focused on monthly, quarterly, and annual data, where weather data from the same month or quarter from previous years were used as training data. This approach aimed to improve the stability of the model by reducing the noise from daily fluctuations and emphasizing more obvious seasonal patterns.

With better results, the experiment was conducted on one weather feature, TX, at one weather station in Indonesia. The YoY mechanism was applied to 192 stations in Indonesia to validate the model performance further. With limited resources, the experiment was conducted on only one feature by forecasting monthly and quarterly using three machine-learning models. The results of the experiment are summarized in Fig. 11 and Table IV.

Based on Table IV, LSTM performs best overall, especially on monthly data with a MAPE value of 0.012744, demonstrating its ability to model complex patterns. LSTM also excels on annual and quarterly data with MAPE values of 0.006747 and 0.010244, respectively, making it the most reliable method for various time scales. SARIMAX, as a statistical model, excels in annual data with a MAPE of 0.008620, showing its superiority in capturing obvious seasonal patterns. However, there is a decrease in performance on monthly data because it is difficult to learn more complex patterns. Meanwhile, RF Regressor can show reasonably consistent results on each time scale, showing the best performance on quarterly data but not as good as LSTM.

LSTM is the most accurate method, especially on quarterly and monthly data. The SARIMAX model is more suitable for annual data with more apparent seasonal patterns. At the same time, RF Regressor can be used as a simpler alternative but is not as accurate as LSTM.

Based on the results obtained, the performance of each model can be seen even though it has almost the same performance. In terms of time scale, the error that occurs on a quarterly scale is much smaller than on a monthly scale. This is because the pattern of value changes on a quarterly scale tends to be more stable than the changes that occur on a monthly scale, which tends to be more volatile.

The results show that the YoY approach significantly impacts forecasting performance. The idea was obtained after the weather data visualization stage. Standard pre-processing processes such as missing value, transformation, and normalization are insufficient to direct the machine to learn. The right approach is needed to obtain optimal results. This is obtained through a good understanding of the data. Data visualization is essential in obtaining ideas to direct the machine to produce optimal solutions, as shown in Fig. 9.

TABLE III. FORECASTING MODEL PERFORMANCE

Method	MAE	RMSE	MAPE
SARIMAX	1.142	1.152	0.037
RF regressor	0.796	1.075	0.026
LSTM	0.861	0.874	0.028

TABLE IV. FORECASTING YEAR-ON-YEAR MODEL PERFORMANCE

Method	Series	MAE	RMSE	MAPE
SARIMAX	Year	0.229	0.088	0.009
	Quartal	0.345	0.187	0.013
	Month	0.422	0.286	0.016
RF regressor	Year	0.268	0.129	0.010
	Quartal	0.342	0.202	0.013
	Month	0.410	0.289	0.015
LSTM	Year	0.178	0.074	0.007
	Quartal	0.278	0.127	0.010
	Month	0.347	0.206	0.013

I. Classification vs. Forecasting

The choice between classification and forecasting depends on the available dataset and the purpose. Classification and forecasting approaches are often used to interpret future conditions, especially weather. Classification categorizes conditions like future weather, while forecasting provides continuous values like temperature or rainfall. Understanding the available data is an essential factor in determining the optimal solution. There is no fair way to compare classification and forecasting, only based on the desired goal. Based on the dataset available in this study, the forecasting approach is a more appropriate step in utilizing the available data. If forced to use the classification approach, the dataset may accurately predict the weather on a broader time scale, such as weekly, by including weather features on certain days.

IV. CONCLUSION

Weather forecasting is difficult due to the complexity of the interactions that occur. A machine learning approach with proper feature engineering and representation is essential to achieve maximum results. The results of classification and forecasting using three generations of machine learning show insignificant differences, so the initial stage of model development, such as feature engineering, is crucial. Model selection must also consider other factors, such as training time, computational efficiency, and ease of implementation. This aspect is essential in cases that require fast predictions. Understanding data is critical to support the proper and optimal use of data, which can be achieved through good dataset visualization. Classification approaches that perform poorly due to overly general features lead to the use of data for forecasting so that the data can have better benefits. In forecasting time series data such as weather predictions, the data collection or observation time scale needs to be considered. Forecasting over a broader period provides higher

accuracy due to generalization while forecasting over a smaller period has the potential to reduce accuracy due to increased noise or data variation. Regular sequential forecasting produces errors of up to 3%. However, changing a broader time period approach to YoY reduces the error to less than 1.6%. Although the model algorithm is essential to produce good accuracy, basic things such as feature engineering, feature selection, and the methods used are also crucial in creating a good model. In addition, data visualization is needed to support a better understanding of the data so that it makes an optimal approach. Further development can focus on using more sophisticated techniques to recognize features and combine various models to improve performance.

Building on the findings of this study, several directions for future research emerge. First, while the current work employs relatively basic machine learning models such as KNN, SVM, and LSTM, future studies could investigate more advanced or hybrid approaches—such as combining deep learning with statistical or ensemble methods—to enhance predictive accuracy. Expanding the dataset in both spatial and temporal resolution, for instance through hourly measurements or integration of multi-source data from satellites, radar, and reanalysis products, could further improve model performance. Moreover, the present feature set focuses primarily on surface-level variables; incorporating atmospheric parameters, remote sensing indicators, and applying advanced feature engineering could better capture complex climatic interactions. The limited richness of the dataset may also contribute to the underperformance of deep learning models, suggesting that data augmentation techniques or transfer learning from global climate datasets could be beneficial. Hybrid algorithms such as LSTM + SARIMAX, CNN-LSTM, or RF + ARIMA should be tested to leverage both temporal dependencies and nonlinear relationships. The small performance gap between models points to possible bottlenecks in feature selection and preprocessing, warranting a more systematic analysis of feature importance. Given that the best classification accuracy achieved is approximately 0.81, further hyperparameter optimization—using Bayesian or evolutionary search methods—could help unlock additional performance gains. Additionally, the exploration of seasonal decomposition techniques, such as STL decomposition, prior to forecasting could improve signal extraction.

ACKNOWLEDGMENT

This work was supported in part by Sebelas Maret University under Grant No. 228/UN27.22/PT.01.03/2023. The experiments in this study were conducted using Python, with research assistant Rahmat Rohmani providing support in executing the program code.

REFERENCES

[1] D. Didiharyono, G. Giarno, and S. Sapareng, "Changes in rainfall intensity, rising air temperature, wind speed, and its relationship with land use in makassar city," *JST (Jurnal Sains dan Teknologi)*, vol. 11, no. 2, p. 441–450, Aug. 2022. [Online]. Available: https://ejournal.undiksha.ac.id/index.php/JST/article/view/45604

[2] M. Anwar, S. Nugrohadi, V. Tantriyati, and V. Windarni, "Rain prediction using rule-based machine learning approach," *Advance Sustainable Science, Engineering and Technology*, vol. 2, no. 1, p. 0200104, 2020. [Online]. Available: http://journal.upgris.ac.id/index.php/asset/article/view/6019

[3] J. Helali, M. Nouri, M. Mohammadi Ghaleni, S. A. Hosseni, F. Safarpour, A. Shirdeli, P. Paymard, and Z. Kalantari, "Forecasting precipitation based on teleconnections using machine learning approaches across different precipitation regimes," *Environmental Earth Sciences*, vol. 82, no. 21, p. 495, Sep 2023. [Online]. Available: https://doi.org/10.1007/s12665-023-11191-9

[4] C. Lalitha and D. Ravindran, "Hybrid deep learning framework for weather forecast with rainfall prediction using weather bigdata analytics," *Multimedia Tools and Applications*, vol. 83, no. 25, pp. 66 413–66 432, Jul 2024. [Online]. Available: https://doi.org/10.1007/s11042-023-17801-9

[5] S. Kumari and P. Muthulakshmi, "Sarima model: An efficient machine learning technique for weather forecasting," *Procedia Comput. Sci.*, vol. 235, no. C, p. 656–670, Jan. 2024. [Online]. Available: https://doi.org/10.1016/j.procs.2024.04.064

[6] L. Chen, B. Han, X. Wang, J. Zhao, W. Yang, and Z. Yang, "Machine learning methods in weather and climate applications: A survey," *Applied Sciences*, vol. 13, no. 21, p. 12019, 2023. [Online]. Available: https://www.mdpi.com/2076-3417/13/21/12019

[7] M. Abdullah and S. Said, "Performance evaluation of machine learning regression models for rainfall prediction," *Iranian Journal of Science and Technology - Transactions of Civil Engineering*, dec 2024. [Online]. Available: https://link.springer.com/10.1007/s40996-024-01691-4

[8] L. Chen, X. Zhong, F. Zhang, Y. Cheng, Y. Xu, Y. Qi, and H. Li, "Fuxi: a cascade machine learning forecasting system for 15-day global weather forecast," *npj Climate and Atmospheric Science*, vol. 6, no. 1, p. 190, Nov 2023. [Online]. Available: https://doi.org/10.1038/s41612-023-00512-1

[9] J. Chen, H. Dai, S. Wang, and C. Liu, "Improving accuracy and efficiency in time series forecasting with an optimized transformer model," *Engineering Letters*, vol. 32, no. 1, p. 1–11, jan 2024.

[10] L. Parviz, K. Rasouli, and A. Torabi Haghighi, "Improving hybrid models for precipitation forecasting by combining nonlinear machine learning methods," *Water Resources Management*, vol. 37, no. 10, pp. 3833–3855, Aug 2023. [Online]. Available: https://doi.org/10.1007/s11269-023-03528-7

[11] T. Tang, T. Liu, and G. Gui, "Forecasting precipitation and temperature evolution patterns under climate change using a random forest approach with seasonal bias correction," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 17, pp. 12 609–12 621, 2024. [Online]. Available: https://doi.org/10.1109/JSTARS.2024.3425639

[12] F. Rezaei Aderyani and S. Jamshid Mousavi, "Machine learning-based rainfall forecasting in real-time optimal operation of urban drainage systems," *Journal of Hydrology*, vol. 645, p. 132118, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0022169424015142

[13] P. S. Saketh, R. Rohit, and B. Suneetha, "Weather forecasting using machine learning," in *2023 International Conference on Inventive Computation Technologies (ICICT)*, 2023, pp. 13–18. [Online]. Available: https://doi.org/10.1109/ICICT57646.2023.10134218

[14] C. Sengoz, S. Ramanna, S. Kehler, R. Goomer, and P. Pries, "Machine learning approaches to improve north american precipitation forecasts," *IEEE Access*, vol. 11, pp. 97 664–97 681, 2023. [Online]. Available: https://doi.org/10.1109/ACCESS.2023.3309054

[15] H. Zhang, Y. Liu, C. Zhang, and N. Li, "Machine learning methods for weather forecasting: A survey," *Atmosphere*, vol. 16, no. 1, p. 82, 2025. [Online]. Available: https://www.mdpi.com/2073-4433/16/1/82

[16] V. Kumar, N. Kedam, O. Kisi, S. Alsulamy, K. M. Khedher, and M. A. Salem, "A comparative study of machine learning models for daily and weekly rainfall forecasting," *Water Resources Management*, vol. 39, no. 1, pp. 271–290, Jan 2025. [Online]. Available: https://doi.org/10.1007/s11269-024-03969-8

[17] K. Chattrairat, W. Wongseree, and A. Leelasantitham, "Comparisons of machine learning methods of statistical downscaling method: Case studies of daily climate anomalies in thailand," *Journal of Web Engineering*, vol. 20, no. 5, p. 1397–1424, Jul. 2021. [Online]. Available: https://journals.riverpublishers.com/index.php/JWE/article/view/4869

[18] M. Akbarian, B. Saghafian, and S. Golian, "Monthly streamflow forecasting by machine learning methods using dynamic weather prediction model outputs over iran," *Journal of Hydrology*, vol. 620, p. 129480, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0022169423004225

[19] M. A. Abed, M. A. Imteaz, and A. N. Ahmed, "Utilising machine learning for pan evaporation prediction - a case study in western australia," in *International MultiConference of Engineers and Computer Scientists 2023 (IMECS 2023)*, vol. 2245, 2023, Conference paper, p. 14 – 18, cited by: 3. [Online]. Available: https://irepository.uniten.edu.my/handle/123456789/34493

[20] D. Maheng, B. Bhattacharya, C. Zevenbergen, and A. Pathirana, "Changing urban temperature and rainfall patterns in jakarta: A comprehensive historical analysis," *Sustainability*, vol. 16, no. 1, p. 350, 2024. [Online]. Available: https://www.mdpi.com/2071-1050/16/1/350

[21] H. Darmawan, M. Yuliana, and M. Z. S. Hadi, "Gru and xgboost performance with hyperparameter tuning using gridsearchcv and bayesian optimization on an iot-based weather prediction system," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 13, pp. 851–862, 6 2023. [Online]. Available: https://doi.org/10.18517/ijaseit.13.3.18377

[22] D. P. Hapsari, M. I. Utoyo, and S. W. Purnami, "A prediction of rainfall data based on support vector machine with stochastic gradient descent," in *Proceedings of the 5th International Workshop on Innovations in Information and Communication Science and Technology*. CEUR-WS, 3 2020. [Online]. Available: https://ceur-ws.org/Vol-2627/short10.pdf

[23] M. T. Anwar, E. Winarno, W. Hadikurniawati, and M. Novita, "Rainfall prediction using extreme gradient boosting," *Journal of Physics: Conference Series*, vol. 1869, p. 12078, 4 2021. [Online]. Available: https://dx.doi.org/10.1088/1742-6596/1869/1/012078

[24] Y. Ruhiat, "Forecasting rainfall and potential for repeated events to predict flood areas in banten province, indonesia," *Journal of Measurements in Engineering*, vol. 10, pp. 68–80, 6 2022. [Online]. Available: https://doi.org/10.21595/jme.2022.22363

[25] B. Prihasto, D. Darmansyah, D. P. Yuda, F. M. Alwafi, H. N. Ekawati, and Y. P. Sari, "Comparative analysis of k-means and k-medoids clustering methods on weather data of denpasar city," *Jurnal Pendidikan Multimedia (Edsence)*, vol. 5, pp. 91–114, 12 2023. [Online]. Available: https://doi.org/10.17509/edsence.v5i2.65925

[26] A. L. R. Putri, "Performance analysis of the support vector machine algorithm in predicting rain potential in dki jakarta," *Journal of Artificial Intelligence and Engineering Applications (JAIEA)*, vol. 3, pp. 671–676, 6 2024. [Online]. Available: https://doi.org/10.59934/jaiea.v3i3.490

[27] A. A. Akbar, Y. Darmawan, A. Wibowo, and H. K. Rahmat, "Accuracy assessment of monthly rainfall predictions using seasonal arima and long short-term memory (lstm)," *Journal of Computer Science and Engineering (JCSE)*, vol. 5, pp. 99–114, 8 2024. [Online]. Available: https://icsejournal.com/index.php/JCSE/article/view/843

[28] M. Putra, M. S. Rosid, and D. Handoko, "High-resolution rainfall estimation using ensemble learning techniques and multisensor data integration," *Sensors*, vol. 24, no. 15, p. 5030, 2024. [Online]. Available: https://www.mdpi.com/1424-8220/24/15/5030

[29] V. Monita, S. Raniprima, and N. Cahyadi, "Comparative analysis of daily and weekly heavy rain prediction using lstm and cloud data," *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, vol. 10, pp. 833–842, 12 2024. [Online]. Available: https://journal.uad.ac.id/index.php/JITEKI/article/view/30374

[30] F. Simanjuntak, I. Jamaluddin, T.-H. Lin, H. A. W. Siahaan, and Y.-N. Chen, "Rainfall forecast using machine learning with high spatiotemporal satellite imagery every 10 minutes," *Remote Sensing*, vol. 14, no. 23, p. 5950, 2022. [Online]. Available: https://www.mdpi.com/2072-4292/14/23/5950

[31] M. F. Sodiq and F. Amin, "Sentiment analysis of bmkg weather information service using k-nearest neighbor method," *International Journal Software Engineering and Computer Science (IJSECS)*, vol. 4, pp. 822–835, 8 2024. [Online]. Available: https://doi.org/10.35870/ijsecs.v4i2.2881

[32] I. M. Murwantara, P. Yugopuspito, and R. Hermawan, "Comparison of machine learning performance for earthquake prediction in indonesia using 30 years historical data," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 18, pp. 1331–1342, 6 2020. [Online]. Available: https://doi.org/10.12928/telkomnika.v18i3.14756

[33] A. G. Salman, B. Kanigoro, and Y. Heryadi, "Weather forecasting using deep learning techniques," in *2015 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 2015, pp. 281–285. [Online]. Available: https://doi.org/10.1109/ICACSIS.2015.7415154

[34] I. D. G. A. Putra, H. Nimiya, A. Sopaheluwakan, T. Kubota, H. S. Lee, R. P. Pradana, M. N. F. Alfata, R. B. Perdana, D. S. Permana, N. F. Riama, and D. Karnawati, "Development of typical meteorological years based on quality control of datasets in indonesia," *Renewable Energy*, vol. 221, p. 119699, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0960148123016142

[35] F. Rahutomo, T. Kitasuka, and M. Aritsugi, "Test collection recycling for semantic text similarity," in *ACM International Conference Proceeding Series*, 2012. [Online]. Available: https://doi.org/10.1145/2428736.2428784

[36] C. Rahmad, F. Rahutomo, M. A. Gustalika, and I. F. Rahmah, "Identification of sugarcane maturity scale based on rgb, gabor feature extraction and support vector machine," *IOP Conference Series: Materials Science and Engineering*, vol. 434, 12 2018. [Online]. Available: https://doi.org/10.1088/1757-899X/434/1/012065

[37] F. Rahutomo, A. S. Mulyo, and P. Y. Saputra, "Automatic grammar checking system for indonesian," in *2018 International Conference on Applied Science and Technology (iCAST)*, 2018, pp. 308–313. [Online]. Available: https://doi.org/10.1109/iCAST1.2018.8751591

[38] A. A. Septarina, F. Rahutomo, and M. Sarosa, "Machine translation of indonesian: a review," *Communications in Science and Technology*, vol. 4, pp. 12–19, 7 2019. [Online]. Available: https://doi.org/10.21924/cst.4.1.2019.104

[39] M. F. S. Thirafi and F. Rahutomo, "Implementation of naïve bayes classifier algorithm to categorize indonesian song lyrics based on age," in *2018 International Conference on Sustainable Information Engineering and Technology (SIET)*, 2018, pp. 106–109. [Online]. Available: https://doi.org/10.1109/SIET.2018.8693201

[40] F. Rahutomo, A. S. Mulyo, and P. Y. Saputra, "Automatic grammar checking system for indonesian," in *2018 International Conference on Applied Science and Technology (iCAST)*, 2018, pp. 308–313. [Online]. Available: https://doi.org/10.1109/iCAST1.2018.8751591

[41] "Climate data daily idn," https://tinyurl.com/bmkgIDN2020.

[42] "Semarang daily climate data 2020-2023," https://tinyurl.com/bmkgSMG2023.

[43] M. M. Hassan, M. A. T. Rony, M. A. R. Khan, M. M. Hassan, F. Yasmin, A. Nag, T. H. Zarin, A. K. Bairagi, S. Alshathri, and W. El-Shafai, "Machine learning-based rainfall prediction: Unveiling insights and forecasting for improved preparedness," *IEEE Access*, vol. 11, pp. 132 196–132 222, 2023. [Online]. Available: https://doi.org/10.1109/ACCESS.2023.3333876

[44] N. Hayaty, H. Kurniawan, M. R. Rathomi, F. Chahyadi, and M. Bettiza, "Rainfall prediction with support vector machines: A case study in tanjungpinang city, indonesia," *BIO Web of Conferences*, vol. 70, p. 1003, nov 2023. [Online]. Available: https://www.bio-conferences.org/10.1051/bioconf/20237001003

[45] C. Yin, L. T. Binh, D. T. Anh, S. T. Mai, A. Le, V.-H. Nguyen, V.-C. Nguyen, N. X. Tinh, H. Tanaka, N. T. Viet, L. D. Nguyen, and T. Q. Duong, "Advanced machine learning techniques for predicting nha trang shorelines," *IEEE Access*, vol. 9, pp. 98 132–98 149, 2021. [Online]. Available: https://doi.org/10.1109/ACCESS.2021.3095339

[46] R. Teixeira, A. Cerveira, E. J. S. Pires, and J. Baptista, "Enhancing weather forecasting integrating lstm and ga," *Applied Sciences*, vol. 14, no. 13, p. 5769, 2024. [Online]. Available: https://www.mdpi.com/2076-3417/14/13/5769

[47] F. R. Alharbi and D. Csala, "A seasonal autoregressive integrated moving average with exogenous factors (sarimax) forecasting model-based time series approach," *Inventions*, vol. 7, no. 4, p. 94, 2022. [Online]. Available: https://www.mdpi.com/2411-5134/7/4/94

[48] T. A. Gahwera, O. S. Eyobu, and M. Isaac, "Analysis of machine learning algorithms for prediction of short-term rainfall amounts using uganda's lake victoria basin weather dataset," *IEEE Access*, vol. 12, pp. 63 361–63 380, 2024. [Online]. Available: https://doi.org/10.1109/ACCESS.2024.3396695

[49] P. St-Aubin and B. Agard, "Precision and reliability of forecasts performance metrics," *Forecasting*, vol. 4, no. 4, pp. 882–903, 2022. [Online]. Available: https://www.mdpi.com/2571-9394/4/4/48

Graph-Based Clustering of Short Texts Using Word Embedding Similarity

Supakpong Jinarat, Ratchakoon Pruengkarn
Information Technology and Artificial Intelligence Program,
College of Engineering and Technology, Dhurakij Pundit University, Bangkok, Thailand 10210

***Abstract*—The exponential growth of short textual content on the Internet, such as social media posts and search snippets, necessitates effective text mining techniques. Short text clustering, a critical tool for organizing this data, contends with two primary challenges: data sparsity, which undermines the quality of traditional clustering methods, and the poor interpretability of machine-generated cluster labels. This study introduces the Semantic Word Graph (SWG) algorithm, a novel graph-based approach designed to address both of these issues simultaneously. Our methodology begins by constructing a global word graph where nodes represent unique terms from the corpus, and edges are weighted by the semantic similarity of word pairs, calculated using a pre-trained Word2Vec model. Cohesive communities of words are then identified using the Louvain method, and documents are assigned to clusters based on these communities. Meaningful cluster labels are generated by ranking representative nouns within each community. To validate our approach, the SWG algorithm was evaluated on three benchmark datasets (AG News, Tweet, and SearchSnippets) and compared against established methods, including Lingo, Suffix Tree Clustering (STC), and K-means. Quantitative results, measured by the F-score, show that SWG achieved up to 0.89 F-score on AG News, 0.85 on Tweets, and 0.82 on SearchSnippets, consistently outperforming baseline algorithms in clustering quality. Furthermore, a qualitative analysis confirms that SWG produces more coherent and topically comprehensive cluster labels, improving interpretability. This study concludes that the SWG algorithm is a robust and effective framework for enhancing both the accuracy and interpretability of short text clustering. Future research could explore integrating contextual embeddings such as BERT to capture deeper semantic relationships, optimizing the similarity threshold dynamically for different datasets, and scaling the algorithm to handle larger, real-time streaming text data. These directions would further improve the applicability of SWG in diverse domains such as social media analytics, news aggregation, and real-time topic detection.**

***Keywords*—Clustering; graph-based clustering; semantic similarity; short text; word embedding**

I. Introduction

Short text clustering is a specialized form of text clustering that focuses on grouping brief textual content—such as tweets, comments, and news headlines—into coherent categories based on their content. Unlike traditional text clustering methods that deal with longer documents, short text clustering faces unique challenges due to data sparsity, informal language usage, and high noise levels. With the exponential growth of short text data on social media platforms, news feeds, and online forums, the demand for accurate and interpretable clustering methods has become increasingly critical.

Current approaches often suffer from degraded clustering quality when dealing with sparse data and have difficulty generating labels that are both meaningful and representative of the underlying topics. These limitations reduce the practical usability of clustering results in real-world applications such as trend detection, news aggregation, and customer feedback analysis. Moreover, while deep learning and embedding-based models have improved text representation, many methods still fail to fully exploit semantic relationships between words in short texts, leading to clusters that lack coherence.

Although several clustering techniques, including graph-based and embedding-enhanced methods, have been proposed, most still struggle with balancing clustering accuracy and label interpretability. Few approaches explicitly address both challenges in a unified framework, and existing solutions often require extensive parameter tuning or rely on domain-specific data, limiting their generalizability.

This study aims to address these challenges by introducing the Semantic Word Graph (SWG) algorithm, a novel graph-based clustering approach designed specifically for short text. The proposed method enhances data representation using pre-trained word embeddings, detects cohesive semantic communities, and generates interpretable cluster labels through part-of-speech-based ranking. By focusing on both clustering quality and label interpretability, this research seeks to deliver a more robust and practical solution for real-world short text analysis.

II. Related Work

Despite the challenges, short text clustering has a wide range of real-world applications. In social media analysis, it is used to identify trending topics [1], [2], [3], [4], [5], track public opinion [6], and detect events [7]. In news aggregation, it helps organize news into structured categories. In customer service, clustering improves response time by categorizing customer inquiries. It is also applied in sentiment analysis to classify short texts into positive, negative, or neutral categories [8], [9].

Recently, neural network-based approaches have been adopted for short text representation learning. For example, [10] proposed a distributional semantic model to improve semantic understanding in clustering, while [11] developed STC2, which applies convolutional neural networks (CNN) [12] to feature learning. STC2 converts short text features into compact binary codes and applies k-means for clustering.

Word embedding is a procedure used in Natural Language Processing (NLP) and machine learning to transform individual words to the numerical value of vector. Word embedding enables computers to analyze and process human language by

transforming words into numerical vectors which reveal some semantic relationships between them.

Word2vec is a specific algorithm to learn word embeddings from large text corpus. It was developed by a team of researchers at Google in 2013 [13]. The algorithm employs a neural network to acquire the contextual associations among words within an extensive text corpus. Neural network is trained to predict a word appearing given its surrounding words. The weights of the neural network are used as the word embeddings.

Word embeddings are powerful because they capture the relationships between words in a language. For example, in a vector space created by word embeddings, words that are semantically similar are located close to each other. This allows computers to perform semantic tasks such as finding synonyms, detecting word relationships, and even predicting the sentiment of text.

The fundamental concept underlying word2vec involves utilizing a neural network to forecast the context words that encompass a target word, leveraging a substantial text corpus. The neural network is trained to encode each word as a vector of real values, ensuring that similar words are positioned close together in the high-dimensional space. This is achieved through a training process called stochastic gradient descent, which updates the weights of the neural network in a way that minimizes the difference between the predicted and actual context words.

There are two primary architectures of word2vec. First, Skip-gram model is to predict context words by using a target word. Secondly, Continuous Bag-of-Words (CBOW) model is to predict the target word based on its context. Both architecture have demonstrated their effectiveness in generating high-quality word embeddings, and the selection between them relies on the task at hand and the corpus size.

III. Methodology

In this research, we introduce a clustering algorithm designed specifically for short text documents. Our algorithm includes six primary steps, as shown in Fig. 1. These steps include pre-processing, construction of a word semantic graph, subgraph detection, document assignment, cluster selection, and cluster label generation. By following this methodology, our aim is to produce high-quality clusters with meaningful results.

A. Preprocessing

Pre-processing is an essential step in text clustering to clean and transform raw text data into a suitable format for clustering algorithms. We remove special characters, punctuation marks, and numbers from the text then eliminate stop words (commonly used words that do not carry significant meaning) such as 'and', 'the', 'is', etc. Irrelevant or noisy elements such as URLs, HTML tags, and email addresses are removed. For Tweet dataset, Twitter-specific elements like hashtags, mentions, and retweet are removed as well.

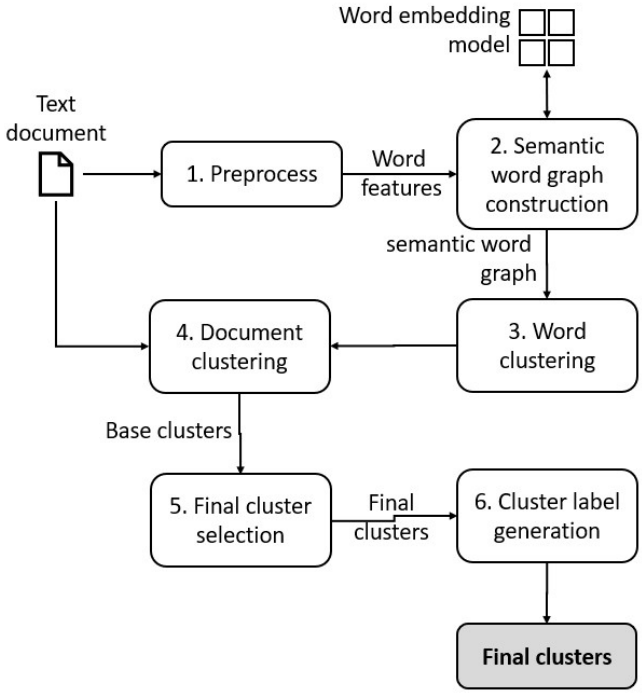


Fig. 1. System architecture of our proposed algorithm.

B. Semantic Word Graph Construction

After preprocessing, a semantic word graph is constructed based on the word features obtained from the previous step. We utilize a pre-trained Word2Vec model trained on the Google News corpus, containing approximately 100 billion words. The model provides 300-dimensional vector representations for approximately three million unique words [14].

The graph is defined as $G = (V, E, W)$, where V is the set of unique words from the preprocessed text. E is the set of undirected edges e_{ij} between word pairs v_i , $v_j \in V$ and W is the set of edge weights w_{ij}, which represent the semantic similarity between words v_i and v_j computed using the cosine similarity of their word vectors.

$$w_{ij} = sim(v_i, v_j) = \frac{v_i \cdot v_j}{\|v_i\|\|v_j\|} \tag{1}$$

Only edges with weights exceeding a predefined threshold (e.g. $w_{ij} > 0.5$) are retained to construct a meaningful semantic network.

D1: "The football team won the championship match with a score of 3-1.",
D2: "The basketball player made an incredible slam dunk during the game.",
D3: "The golfer sank a long putt to secure victory in the golf tournament.",
D4: "The president delivered a speech outlining the administration's goals.",
D5: "The prime minister met with foreign leaders to discuss international relations.",
D6: "The senator proposed a bill to reform the healthcare system."

Fig. 2. Example of text documents used for demonstration of word semantic graph construction.

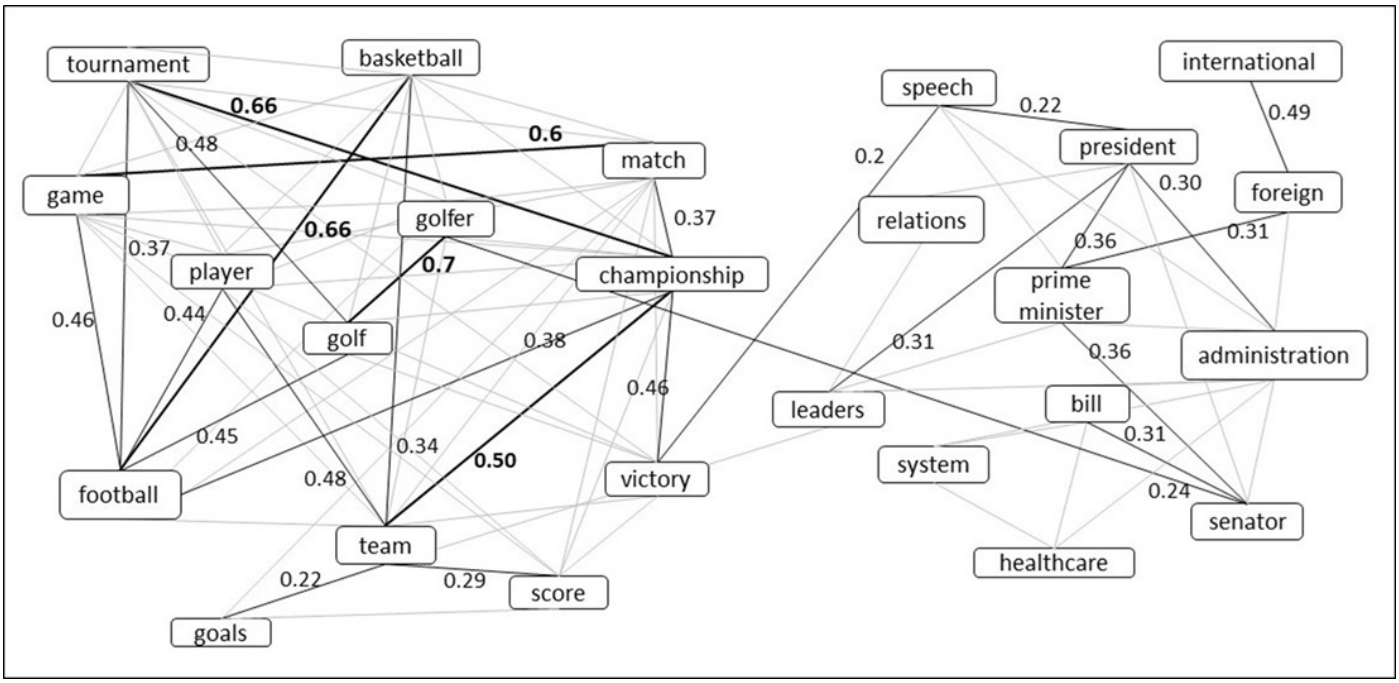


Fig. 3. Illustrative of word semantic graph construction from text document example.

Fig. 3 showcases the construction of the word semantic graph, derived from the example text document illustrated in Fig. 2.

C. Word Clustering

In the semantic word graph produced from the previous step, words with high semantic similarity are connected by edges with weights close to 1. To identify cohesive groups of related words, the graph is divided into subgraphs (or word clusters) by pruning weak edges—those with weights below a predefined similarity threshold st.

After pruning, we apply the Louvain method [15] for community detection to each resulting subgraph. This algorithm detects communities based on modularity optimization, helping identify groups of words that exhibit strong semantic relationships. As shown in Fig. 4, when a similarity threshold of 0.3 is applied, the semantic word graph is partitioned into two distinct subgraphs. Each subgraph contains strongly connected words, which are treated as word clusters. These word clusters serve as the input for the next step: document clustering, where documents are grouped based on their semantic similar derived from these clusters.

Each subgraph produced from this step is treated as a word cluster, with each cluster containing a set of candidate labels—i.e., representative words from the cluster—but no associated documents at this stage. The assignment of documents to these clusters will be operated in the next step of the algorithm.

As illustrated in Fig. 4, two example word clusters are presented.

The first word cluster includes candidate labels such as: “football”, “team”, “championship”, “match”, “score”, “basketball”, “player”, “game”, “tournament”, “golfer”, “victory”, “golf”, and “goals”.

The second word cluster contains labels like: “president”, “administration”, “prime minister”, “foreign”, “leaders”, “international”, “senator”, and “bill”.

These clusters are semantically coherent and form the basis for the upcoming step of document assignment based on semantic matching.

D. Document Clustering

The semantic word subgraphs produced from the previous phase are used as word clusters for the purpose of document clustering. Each document is assigned to one or more clusters based on the presence of any word from the set of candidate labels within the document text. This approach allows a document to be associated with multiple word clusters, resulting in overlapping clustering. Such flexibility is useful in cases where a document covers multiple topics or contains ambiguous content. The resulting groups of documents—each associated with a semantic word cluster—are referred to as base clusters, which used as input for the next stages of the clustering framework.

E. Final Cluster Selection

In the previous step, we define the importance of a base cluster based on its size—that is, the number of documents it contains. Accordingly, we rank all base clusters in descending order of size. The top k clusters in this ranking are selected as the final output clusters for further interpretation and evaluation.

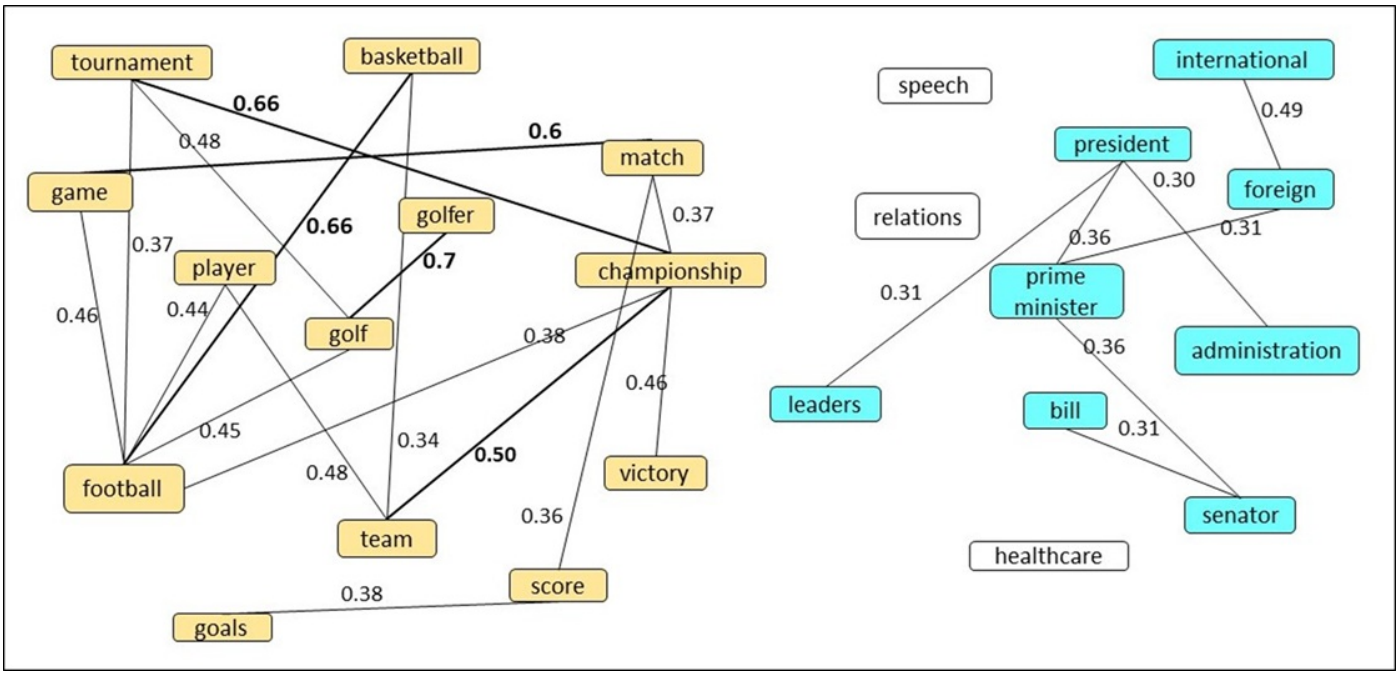


Fig. 4. An illustrative instance of a word semantic graph and its corresponding subgraphs is presented, utilizing a similarity threshold of 0.3. The semantic word graph displays the interconnections among words, while the subgraphs depict clusters formed based on the established similarity threshold.

F. Generating Cluster Label

Beyond clustering efficiency, it is equally essential to provide a descriptive interpretation of each resulting cluster. In this step, we focus on generating cluster labels for the final output clusters. As each semantic subgraph represents a set of semantically related words, we consider these words as candidate descriptors for the associated documents. Specifically, we first eliminate non-noun words using part-of-speech filtering. Next, each word is assigned a relevance score computed using a scoring function [see Eq. (2)], which quantifies its representativeness.

$$label_score(f_i) = df(f_i) \times neighbor(v_i) \tag{2}$$

Where $df(f_i)$ is document frequency of word f_i from the input text document and neighbor(vi) is number of nodes that links to the node v_i in the semantic word graph.

Finally, the top-ranking words—typically those with the highest scores—are selected as cluster labels, offering an interpretable summary of each semantic group.

IV. Experiments

A. Experimental Setup

We evaluated the performance of our proposed algorithm, Semantic Word Graph (SWG), by comparing it with established text clustering methods, including Lingo [16], Suffix Tree Clustering (STC) [17], [18] and the K-means clustering algorithm. For both Lingo and STC, default parameter settings recommended by their original implementations were adopted.

For SWG, we experimented with three different values of the similarity threshold: $st = 0.4$, $st = 0.5$, and $st = 0.6$. The performance of all algorithms was assessed across a range of cluster counts, from 1 to 10, which aligns with the actual number of topics in each dataset used.

B. Dataset

The experiments were conducted on three widely used benchmark datasets for short text clustering: namely the SearchSnippets, Tweet, and AG News datasets.

For the SearchSnippets dataset, which was originally compiled by [19], we used a collection of 12,340 short text snippets retrieved from web search results. These snippets were obtained using predefined keywords associated with 8 distinct topical categories. A summary of this dataset is shown in Table I.

TABLE I. The 8 Different Topics SearchSnippets Dataset

Topic	Number of Documents
Business	1,500
Computers	1,500
Culture-Arts-Entertainment	2,210
Education-Science	2,660
Engineering	370
Health	1,180
Politics-Society	1,500
Sports	1,420

Secondly, for the Tweet dataset, we collected tweets via web scraping from 16 prominent public Twitter accounts, focusing on 4 specific topics, as summarized in Table II. A maximum of 2,000 tweets was retrieved from each account on 14 May 2022. The retrieved tweets were categorized according

to their corresponding topics. Subsequently, 2,000 tweets were randomly selected for each topic, resulting in a final dataset of 8,000 tweets (across four topics) used in our experimental analysis

TABLE II. THE TWITTER ACCOUNTS WERE CATEGORIZED INTO 4 DISTINCT TWEET TOPICS, ENSURING SEPARATION BASED ON THEIR CONTENT

Politics	Sports	Sci/Tech	Business
BBC_Politics	ESPN	NASA	CNNBusiness
CNN_Politics	Fox_Sports	Science_News	Davos
POLITICO	SportsCenter	science	economics
Post_Politics	Twitter_Sports	CERN	EconomicTimes

Lastly, we used the AG News dataset, a large corpus of online news articles commonly employed in text classification and clustering tasks [20]. This dataset contains 496,835 news articles collected from over 2,000 distinct news outlets. For our experiments, we randomly selected 1,000 articles from each six categories, resulting in a total of 6,000 documents. Only the title and description fields were used for clustering. A summary of this dataset is presented in Table III.

TABLE III. AG NEWS DATASET SEPARATED BY 6 DIFFERENT TOPICS

Topic	Number of Documents
Business	1,000
Entertainment	1,000
Health	1,000
Sci/Tech	1,000
Software and Development	1,000
Sports	1,000

C. Evaluation Method

We evaluated the effectiveness of our proposed clustering algorithm by measuring its ability to accurately group short texts and address the inherent challenges of short-text data. Experiments were conducted on multiple benchmark datasets consisting of short text documents. Our method was compared with established clustering techniques, including Lingo and Suffix Tree Clustering (STC). The primary evaluation metric used was the F-score.

The F-score is a widely used evaluation metric in machine learning, particularly for classification and clustering tasks. It represents the harmonic mean of precision and recall, thereby providing a balanced measure of accuracy and completeness.

Precision is the proportion of true positive predictions among all positive predictions made by the model. It quantifies the accuracy of positive assignments.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{3}$$

Recall is the proportion of true positive predictions among all actual positive instances. It reflects the completeness of the positive predictions.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{4}$$

The F-score can be computed using two standard variants: micro-averaging and macro-averaging. The distinction between these methods lies in how precision and recall are aggregated across multiple clusters.

The micro-average F-score [21], [22], [23], [24] computes performance globally by counting the total true positives, false positives, and false negatives across all clusters. It treats the entire clustering task as a single binary classification problem, where documents that belong to any cluster are considered the positive class. This approach is particularly useful when class imbalance exists, as it emphasizes frequent categories. The micro-average F-score is computed using the following formula:

$$F_{\text{micro}} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5}$$

In contrast, the macro-average F-score computes the F-score for each cluster individually and then averages the results across all clusters. In this case, each cluster is treated as a separate binary classification task, allowing equal weight to be assigned to both large and small clusters. The macro-average F-score is calculated using the following formula:

$$F_i = 2 \times \frac{\text{Precision}_i \times \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i} \tag{5}$$

where Precisioni and Recalli are computed for the i-th cluster. The F-score macro-average [25], [26] is then computed as the average of the F-scores for all clusters:

$$F_{\text{macro}} = \frac{\sum F_i}{n} \tag{7}$$

where n is the total number of clusters.

Moreover, we propose a modification to the traditional macro-average F-score, referred to as the weighted macro-average F-score. This metric considers the size of each cluster when computing the average, thus giving more weight to larger clusters. The weighted macro-average F-score is defined as follows:

$$F_{\text{weighted_macro}} = \frac{\sum(F_i \times n_i)}{N} \tag{8}$$

where n is the total number of documents, F_i denotes the F-score of the $i-th$ cluster, and n_i represents the number of documents in cluster. The weighted macro-average F-score offers a more representative evaluation of clustering performance, as it accounts for the varying sizes of clusters and emphasizes the contribution of each cluster to the overall clustering quality.

V. Result and Discussion

A. The Quality of Clustering

To ensure a fair comparison among Lingo, STC, and our proposed method (SWG), we evaluated all algorithms based on the top-k clusters, where k ranges from 2 to 10. This aligns with the actual number of clusters present in each dataset used in the experiments.

The results clearly show that the proposed SWG algorithm consistently outperformed the baseline methods, particularly on the Tweet dataset, as illustrated in Fig. 6. These findings highlight the significant advantage of incorporating semantic word graphs, especially when dealing with short texts that contain sparse and noisy information.

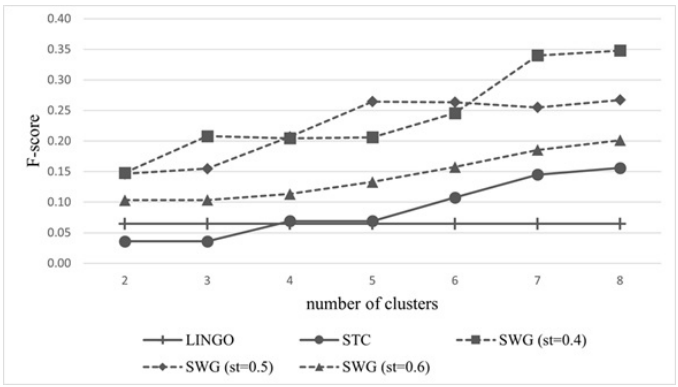


Fig. 5. Quality of clustering on the AG news dataset.

For the AG News dataset (Fig. 5), our method achieved the highest F-score across all tested similarity thresholds, demonstrating its effectiveness in grouping short news articles based on semantic similarity.

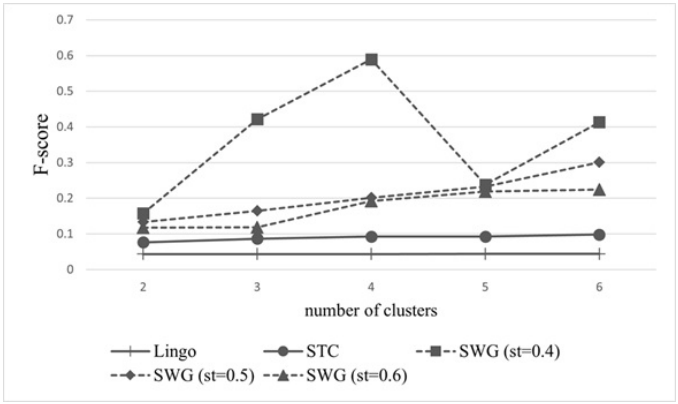


Fig. 6. Quality of clustering on the tweet dataset.

Likewise, on the Tweet dataset, SWG achieved the best F-score under all evaluated threshold values, confirming its robustness in handling short, informal, and noisy text typical of social media content. These results underscore the practical advantage of semantic-based clustering in real-world applications involving brief and informal textual data.

As illustrated in Fig. 7, our proposed method achieved higher F-scores than the baseline algorithms at two similarity

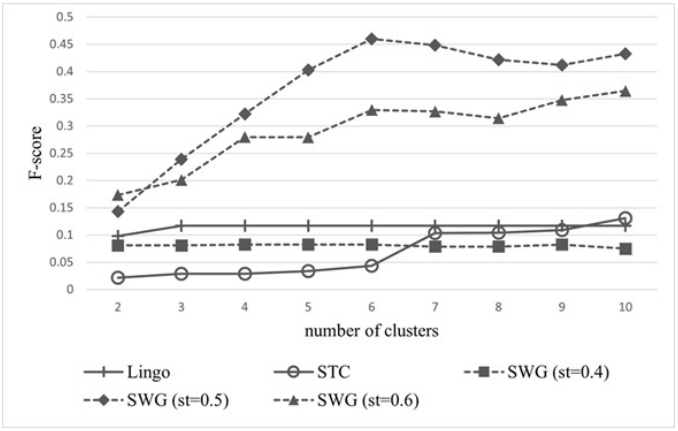


Fig. 7. Quality of clustering on the search snippets dataset.

threshold values ($st = 0.5$ and $st = 0.6$). However, the result for the remaining threshold ($st = 0.4$) was slightly lower than those obtained by the comparative methods.

Despite this, our method still demonstrated competitive performance in clustering short texts from search result snippets, which are often fragmentary and challenging to group semantically. These results further support the robustness of the proposed algorithm across different datasets and parameter settings.

In this study, we also compared the performance of our graph-based text clustering algorithm (SWG) with the well-known vector-based baseline, K-means.

As shown in Fig. 8 and Fig. 10, the SWG algorithm consistently outperformed K-means across all tested cluster numbers on both the AG News and SearchSnippets datasets.

For the Tweet dataset (Fig. 9), our algorithm exhibited slightly lower performance than K-means when the number of clusters was set to 2 and 3. However, SWG outperformed K-means for cluster numbers 4, 5, and 6—most notably when $k = 4$, which corresponds to the ground truth number of clusters in the Tweet dataset. This result demonstrates the strength of SWG in capturing semantically coherent clusters when aligned with real-world topic distributions.

B. Representative Words for a Cluster Result

In this section, we analyze the representative words generated by the three clustering algorithms—Lingo, STC, and SWG—across all three datasets. These representative words reflect the dominant themes within each cluster and allow for qualitative comparisons between the algorithms.

Table IV presents the representative words produced by each algorithm on the AG News dataset, which includes six primary categories: Business, Entertainment, Health, Science/Technology, Software and Development and Sports.

The Lingo algorithm captured Sports-related topics in Cluster 2, 4, and 6 with words like Wins, Wins Gold, and Phelps Wins. It also identified Health-related topics in Cluster 1 and 5 with terms such as Drug and Cancer Drug.

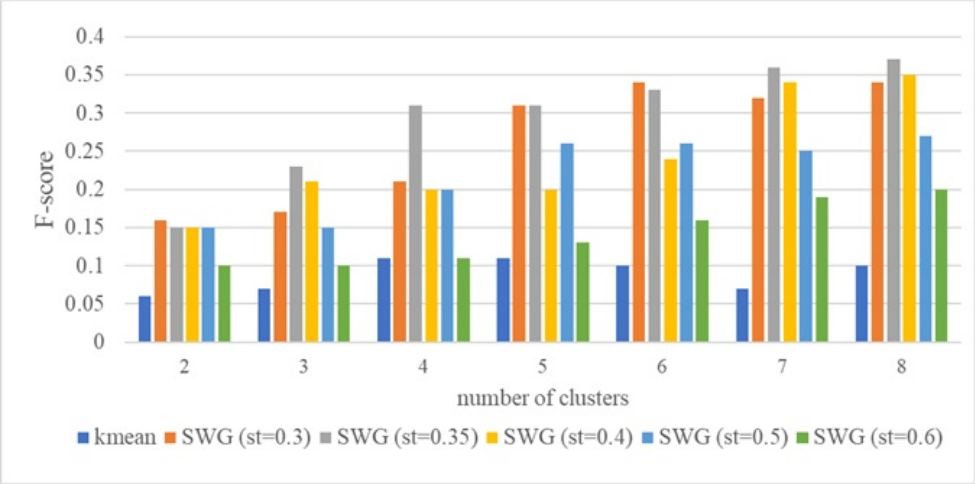


Fig. 8. Quality of clustering of our algorithm compared with k-means algorithm on AG news dataset.

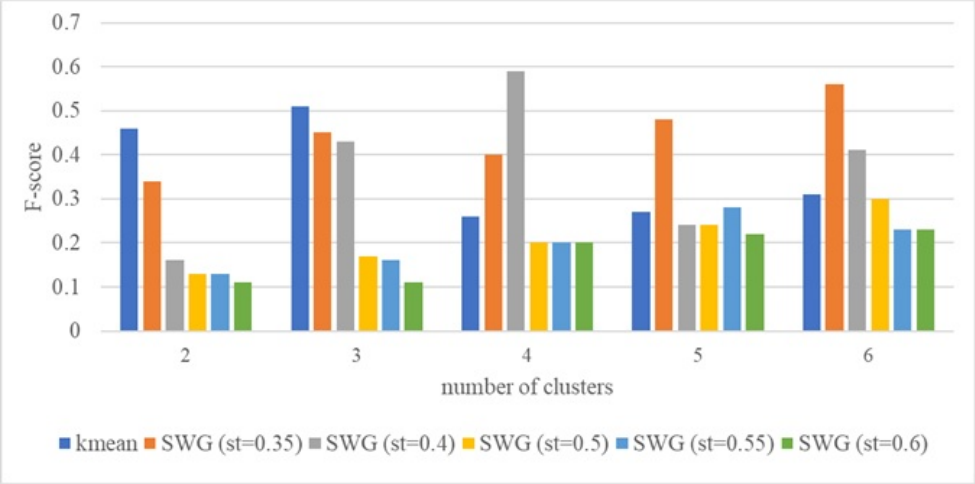


Fig. 9. Quality of clustering of our algorithm compared with k-means algorithm on tweet dataset.

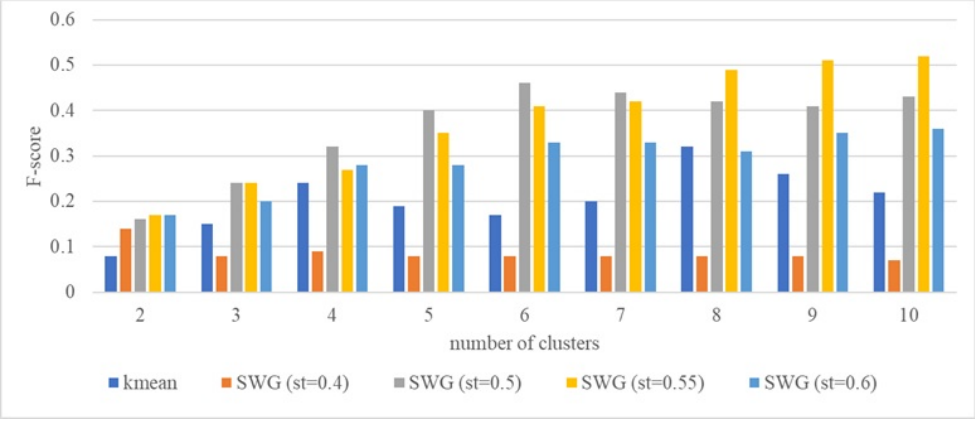


Fig. 10. Quality of clustering of our algorithm compared with k-means algorithm on searchsnippets dataset.

The STC algorithm produced more mixed clusters, capturing terms related to Health Drug, Sports Wins, Business Prices, and Technology Google.

Our proposed method, SWG, demonstrated a broader and more topic-specific range:

Cluster#1: Software and Development — “Microsoft”,

TABLE IV. REPRESENTATIVE WORDS OF CLUSTERS PRODUCED BY EACH COMPETITIVE ALGORITHM ON AG NEWS DATASET

Cluster No.	Lingo	STC	SWG
1	Drug	Drug	Microsoft, Google, IBM, Linux, Windows, Intel
2	Wins	Wins	China, UK, Olympic, US, Sox, Phelps
3	U.S.	U.S.	Profit, Stocks, Prices, Oil, Costs, Trial
4	Wins Gold	Prices	Cancer, Study, Drug, Risk, Health, Disease
5	Cancer Drug	Google	Help, gets, takes, Slow, unveils, Higher
6	Phelps Wins	Oil	Wins, Gets, Takes, Says, Finds, Helps

TABLE V. REPRESENTATIVE WORDS OF CLUSTERS PRODUCED BY EACH COMPETITIVE ALGORITHM ON TWEET DATASET

Cluster No.	Lingo	STC	SWG
1	Boris Johnson Says	Year	people, company, secretary, workers, women, economic, children, business
2	PM Says	Boris Johnson	win, NBA, games, Lakers, team, playoffs
3	Labour Says	LIVE	Labour, MP, UK, Conservative, government, Boris
4	MP Says	Launch	spacecraft, NASA, Earth, astronauts, Mars, science

TABLE VI. REPRESENTATIVE WORDS OF CLUSTERS PRODUCED BY EACH COMPETITIVE ALGORITHM ON SEARCHSNIPPETS DATASET

Cluster No.	Lingo	STC	SWG
1	News News	Edu	research, university, students, school, theory, physics
2	Research Research	News	science, disease, health, computer, biology
3	Wikipedia Wikipedia	Research	web, online, articles, photos, website, art
4	Edu Research	Wikipedia Wiki, Wikipedia Encyclopedia	world, political, gov, games, tournament, election
5	Information News	Gov	movie, film, music, books, band, guitar
6	News Sports	Articles	buy, companies, market, department, industry, organizations
7	Articles News	Computer	linux, wiki, microsoft, cpu, unix, ibm
8	Information Research	Resources	sports, football, soccer, tennis, basketball, president

"Google", "IBM", "Linux", "Windows", "Intel"

Cluster#2: Sports — "China", "UK", "Olympic", "US", "Sox", "Phelps"

Cluster#3: Business — "Profit", "Stocks", "Prices", "Oil", "Costs", "Trial"

Cluster#4: Health — "Cancer", "Study", "Drug", "Risk", "Health", "Disease"

Clusters#5 and Cluster#6: Contained more general and ambiguous terms, making topic identification less conclusive.

These results indicate that the SWG algorithm is more effective at generating semantically coherent cluster labels compared to the baseline algorithms, particularly in identifying multi-category topics in the AG News dataset.

For the results on the Tweet dataset, as presented in Table V, the dataset included four predefined topics: $Politics$, $Business$, $Science/Technology$, and $Sports$.

The Lingo algorithm focused almost exclusively on the Politics topic across all clusters, lacking clear differentiation between other topical categories.

The STC algorithm captured the Politics topic in Cluster#2, while the remaining clusters (Clusters#1, Cluster#3, and Cluster#4) contained more ambiguous or generic terms, such as "Year" "LIVE" and "Launch" which made topic interpretation less definitive.

In contrast, our proposed SWG algorithm successfully identified three distinct topics. Table VI shows respresentarive words of clusters produced by each competitive algorithm on searchSnippets dataset.

VI. CONCLUSION

In this study, we focused on short text clustering with two main objectives: providing high-quality results by grouping together short texts with similar content and generating meaningful cluster labels to aid user understanding.

From the results in section of Quality of Clustering, our proposed algorithm outperformed competitive algorithms across various key parameter variations. The Semantic Word Graph consistently produced significantly better clustering quality.

Furthermore, the representative words generated by the Semantic Word Graph algorithm demonstrated its ability to appropriately represent topics in a cluster. These words struck a balance between being overly specific and excessively general. The cluster labels derived from the Semantic Word Graph algorithm covered multiple topics within the initial clusters, as evidenced by the results in the Result and Discussion section.

The sensitivity analysis conducted on our proposed algorithm offered valuable insights into its performance under different parameter settings. Specifically, by varying the similarity threshold and observing the resulting clustering output, we

gained a deeper understanding of the algorithm's response to input variations.

We observed that increasing the similarity threshold within the subgraph detection step improved the algorithm's performance until an optimal threshold was reached. However, setting the similarity threshold too high led to a slight decrease in clustering performance. Conversely, reducing the similarity threshold resulted in more general semantic connections and an increase in the number of documents per base cluster, thereby impacting the clustering performance. Consequently, determining an appropriate value for the similarity threshold is critical and should be tailored to the specific application and characteristics of the input documents.

In conclusion, our study demonstrated the effectiveness of the proposed Semantic Word Graph algorithm in achieving high-quality clustering results and generating meaningful cluster labels for short texts. The sensitivity analysis provided insights into parameter settings, further enhancing the algorithm's performance and customization capabilities.

Future research could focus on enhancing the SWG framework by integrating advanced contextual embeddings, such as BERT or Sentence-BERT, to capture richer and more nuanced semantic relationships between words. In addition, developing adaptive methods for automatically determining optimal similarity thresholds would allow the algorithm to adjust dynamically to the characteristics of different datasets, improving clustering quality without extensive manual tuning. Finally, incorporating user-centric evaluation—through usability studies and real-world testing—would provide insights into how well the generated cluster labels support human understanding and decision-making, ensuring that the method is not only effective in quantitative terms but also practical and interpretable for end users.

REFERENCES

[1] P. Gurung and R. Wagh, "A study on topic identification using k means clustering algorithm: Big vs. small documents," *Advances in Computational Sciences and Technology*, vol. 10, no. 2, pp. 221–233, 2017.

[2] C. Clifton and R. Cooley, "Topcat: Data mining for topic identification in a text corpus," in *Proc. PKDD 1999: Principles of Data Mining and Knowledge Discovery*, 1999, pp. 174–183.

[3] Z. Ghaemi and M. Farnaghi, "Event detection from geotagged tweets considering spatial autocorrelation and heterogeneity," *Journal of Spatial Science*, vol. 66, no. 3, pp. 353–371, 2021.

[4] J. Weng and B. Lee, "Event detection in twitter," in *Proceedings of the AAAI 2011 International Conference on Web and Social Media*, 2011, pp. 401–408.

[5] S. Behpour, M. Mohammadi, M. V. Albert, Z. S. Alam, L. Wang, and T. Xiao, "Automatic trend detection: Time-biased document clustering," *Knowledge-Based Systems*, vol. 220, p. 106907, 2021.

[6] X. Chen, S. Duan, and L. Wang, "Research on clustering analysis of internet public opinion," *Cluster Computing*, vol. 22, pp. 5997–6007, 2019.

[7] S. B. Kaleel and A. Abhari, "Cluster-discovery of twitter messages for event detection and trending," *Journal of Computational Science*, vol. 6, no. 1, pp. 47–57, 2015.

[8] I. Beregovskaya and M. Koroteev, "Review of clustering-based recommender systems," arXiv preprint arXiv:1705.06157, 2017, available from: https://arxiv.org/abs/1705.06157.

[9] Y. Betancourt and S. Ilarri, "Use of text mining techniques for recommender systems," in *Proceedings of the International Conference on Enterprise Information Systems (ICEIS)*, 2020, pp. 780–787.

[10] M. Kozlowski and H. Rybinski, "Clustering of semantically enriched short texts," *Journal of Intelligent Information Systems*, vol. 53, pp. 69–92, 2018.

[11] J. Xu, B. Xu, P. Wang, P. Zheng, S. Tian, and J. Zhao, "Self-taught convolutional neural networks for short text clustering," *Neural Networks*, vol. 88, pp. 22–31, 2017.

[12] Y. Kim, "Convolutional neural networks for sentence classification," arXiv preprint arXiv:1408.5882, 2022, available from: https://arxiv.org/abs/1408.5882.

[13] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proceedings of the Neural Information Processing Systems Conference (NeurIPS)*, 2013, pp. 3111–3119.

[14] Google Inc., "Google code archive," https://code.google.com/archive/p/word2vec/, 2025, [Online; cited 2025 Mar 12].

[15] V. D. Blondel, J. L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. P10008, 2008.

[16] S. Osiński, J. Stefanowski, and D. Weiss, "Lingo: Search results clustering algorithm based on singular value decomposition," in *Intelligent Information Processing and Web Mining*, ser. Advances in Soft Computing, 2004, vol. 25, pp. 359–368.

[17] O. Zamir and O. Etzioni, "Web document clustering: A feasibility demonstration," in *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, 1998, pp. 46–54.

[18] O. Etzioni and O. Zamir, "Grouper: A dynamic clustering interface to web search results," in *Proceedings of the 8th International Conference on World Wide Web*, 1999, pp. 1361–1374.

[19] X. Phan, L. Nguyen, and S. Horiguchi, "Learning to classify short and sparse text and web with hidden topics from large-scale data collections," in *Proceedings of the 17th International Conference on World Wide Web*, 2008, pp. 91–100.

[20] Y. Tay, M. Dehghani, J. P. Gupta, V. Aribandi, D. Bahri, Z. Qin, and et al., "Are pre-trained convolutions better than pre-trained transformers?" in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, 2021, pp. 4349–4359.

[21] U. Scaiella, P. Ferragina, A. Marino, and M. Ciaramita, "Topical clustering of search results," in *Proceedings of WSDM 2012: International Conference on Web Search and Data Mining*, 2012, pp. 223–232.

[22] M. Yuan, P. Lin, and J. Zobel, "Document clustering vs topic models: A case study," in *Proceedings of the 25th Australasian Document Computing Symposium*, 2021, pp. 1–8.

[23] K. K. Bharti and P. K. Singh, "Hybrid dimension reduction by integrating feature selection with feature extraction method for text clustering," *Expert Systems with Applications*, vol. 42, pp. 3105–3114, 2015.

[24] Y. Fan, L. Shi, and L. Yuan, "Topic modeling methods for short texts: A survey," *Journal of Intelligent & Fuzzy Systems*, vol. 45, pp. 1971–1990, 2023.

[25] D. Crabtree, X. Gao, and P. Andreae, "Improving web clustering by cluster selection," in *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, 2005, pp. 172–178.

[26] S. Lazemi, H. Ebrahimpour-Komleh, and N. Noroozi, "Improving persian dependency-based semantic role labeling using semantic and structural relations," in *Proceedings of the 4th International Conference on Pattern Recognition and Image Analysis*, 2019, pp. 163–167.

A Prompt-Driven Framework for Reflective Evaluation of Course Alignment Using Large Language Models

Mashael M. Alsulami
Department of Information Technology, College of Computers and Information Technology,
Taif University, Taif 21944, Saudi Arabia

***Abstract*—Large language models (LLMs) such as ChatGPT are gaining attention in educational settings, yet their potential role in supporting course design and academic quality assurance remains underexplored. This study introduces a structured, prompt-driven framework that uses ChatGPT as a reflective tool to help faculty and curriculum designers evaluate the alignment between course learning outcomes (CLOs), assessment methods, and teaching strategies. Grounded in the standards of the National Center for Academic Accreditation and Evaluation (NCAAA) and Bloom's Revised Taxonomy, the system generates targeted, context-aware feedback using structured prompts modeled after official NCAAA forms. To enhance reliability and reduce hallucinations, the framework employs template-based prompt engineering and rule-based cognitive classification. A large-scale analysis was conducted across 56 CLOs to assess internal consistency, followed by expert validation from two academic reviewers who evaluated a sample of AI-generated feedback for accuracy, usefulness, and cognitive alignment. The findings highlight the tool's ability to surface alignment issues and offer constructive recommendations, while also demonstrating its potential as a scalable support system for curriculum review and accreditation readiness, complementing rather than replacing human expertise.**

***Keywords*—Prompt engineering; Course Learning Outcomes (CLOs); Large Language Models (LLMs); curriculum alignment; educational quality assurance**

I. Introduction

Ensuring that course design is pedagogically sound and aligned with learning expectations is a central concern for quality assurance in higher education. Curriculum alignment, linking learning outcomes with teaching strategies and assessments, is widely recognized as a core principle of effective instructional design [1]. However, despite the availability of structured accreditation frameworks, several studies have shown that instructors often struggle with articulating measurable outcomes and ensuring that course activities target the appropriate cognitive levels [2][3]. These challenges become more pressing as institutions are required to demonstrate evidence of alignment for accreditation and continuous improvement.

Recent innovations in educational technology have opened new possibilities for automating parts of the curriculum review process. Artificial intelligence (AI), particularly large language models (LLMs), is being explored for its potential to support outcome-based design by providing reflective feedback and consistency checks [4]. These models can help academic staff navigate regulatory requirements, avoid cognitive mismatches between CLOs and assessments, and generate suggestions based on taxonomic alignment principles such as Bloom's taxonomy. However, such tools are not meant to replace human judgment; instead, they offer a structured augmentation to facilitate deeper reflection and compliance with institutional standards.

Creating well-designed courses that truly align learning outcomes with assessments and teaching strategies is essential for delivering quality education, and for meeting accreditation standards. In Saudi Arabia, the National Center for Academic Accreditation and Evaluation (NCAAA) [5] provides clear frameworks and structured forms to help ensure consistency and excellence across higher education institutions. Still, many instructors and curriculum developers find it challenging to interpret these guidelines, detect misalignments, or clearly justify their course design decisions.

This research presents a new approach that brings artificial intelligence into the heart of course evaluation. We introduce a reflective tool powered by ChatGPT, a cutting-edge large language model, to assist faculty in reviewing and improving how CLOs, assessments, and teaching methods are connected. But unlike typical AI applications that generate open-ended content, our tool follows a structured, prompt-driven framework that mirrors the official NCAAA specification format.

What sets this work apart is its contribution to both practice and process: it offers a reliable, transparent, and scalable method for evaluating course alignment. Using Bloom's taxonomy as its foundation, the system identifies the cognitive level of each CLO and checks whether it's well-matched with the stated assessments and teaching strategies. The prompts are carefully designed to guide the model's responses, minimizing hallucination and keeping the feedback focused and consistent.

Rather than replacing human decision-making, this tool augments academic judgment by making the review process more transparent, efficient, and pedagogically sound. It offers a practical contribution to educational quality assurance practices, particularly in accreditation-driven contexts, and opens new directions for human-AI collaboration in curriculum evaluation.

II. Related Work

The integration of artificial intelligence (AI) into higher education has drawn significant attention, particularly with

the emergence of large language models (LLMs) such as ChatGPT. These tools have sparked interest as assistive agents in instructional design, assessment, and academic support. This section reviews existing research on the use of LLMs in educational quality assurance, with a focus on their applications in curriculum evaluation, limitations, and reflective learning support.

A. LLMs in Curriculum and Instructional Design

Several studies have investigated how ChatGPT and similar LLMs can support the instructional design process. For instance, Johansson et al. (2024)[6] explored the ability of ChatGPT to independently design higher education field courses, finding that while the model generated coherent learning objectives and activities, it often failed to account for contextual factors and discipline-specific pedagogy. Similarly, the CDIO 2024 Proceedings [7] include a set of experiments in which ChatGPT was tasked with drafting parts of course specifications. While the model performed well in identifying general structure and terminology, it struggled with generating outcomes that align with the cognitive demands of the discipline.

In line with this, Iftene et al. (2024) [8] emphasized the utility of ChatGPT in supporting educators in the process of formulating learning outcomes. Their findings highlight the model's utility in producing syntactically correct CLOs that often reflect Bloom's Taxonomy, though subject-matter specificity remained a concern.

B. ChatGPT for Assessment Evaluation and Feedback

The potential of ChatGPT in generating or evaluating assessments has also been documented. In a recent study by Van Dis et al. (2024)[9], ChatGPT was tested for its ability to assess the quality of student writing and exams. While it demonstrated promising consistency, its performance deteriorated on open-ended questions where nuanced human judgment was needed. Likewise, Sahlgren et al. (2023)[10] conducted controlled evaluations and concluded that while LLMs can surface generic feedback for rubric-based grading, they are less effective in subjective judgment or when factual correctness is crucial.

C. Limitations and Hallucination Risks

Several works have acknowledged inherent risks in relying on LLMs for educational judgment. Hallucination, generating incorrect or fabricated content, remains a key challenge. Salam. (2023) [11] specifically caution that ChatGPT often invents plausible but inaccurate content, especially in contexts involving institutional policies or accreditation standards. To mitigate this, recent approaches suggest structured prompt engineering [12], limiting model creativity by embedding scaffolding within the input prompt, and validating outputs against expert reviews.

Our approach,relying on structured, template-based prompts that mirror the NCAAA course specification format,builds upon this literature by explicitly controlling the scope of generative output. Additionally, our system incorporates Bloom's taxonomy verb parsing, a method aligned with existing literature [13], to establish a foundation for cognitive demand classification before invoking ChatGPT. This layered methodology aims to reduce drift and hallucination by constraining the model's interpretive boundaries.

D. Reflective Tools and Human-AI Collaboration

While many studies focus on automation, our research contributes to a growing body of work that views LLMs as reflective tools—not replacements for human judgment but catalysts for deeper reflection and review. This aligns with the philosophy of human-AI collaboration articulated by Luo et al. (2023) [14], who argue for maintaining human oversight in contexts where contextual, ethical, or pedagogical nuances are critical.

Our reflective framework differs from fully automated assessment systems by situating the model as a reviewer-in-dialogue rather than an oracle. It provides feedback on the alignment between learning outcomes, teaching strategies, and assessments. Such granular critique, when presented through a dashboard or structured feedback format, facilitates human deliberation and supports transparency in course design decisions.

In summary, our research addresses several limitations found in prior work: (1) it avoids open-ended generation by using grounded prompts and taxonomy-driven structure, (2) it evaluates assessment alignment with cognitive levels rather than generating them, and (3) it augments human judgment rather than substituting it. These features position our reflective tool as a practical and scalable solution for academic quality assurance, particularly in contexts governed by formal standards such as those of the NCAAA in Saudi Arabia.

III. Methodology

This study adopted a computational content analysis approach to evaluate the alignment between Course Learning Outcomes (CLOs) and their corresponding assessment methods within course specification documents. The methodology integrated automated text extraction, verb-level classification [15] using Bloom's Revised Taxonomy [16], and interpretation via a large language model (LLM), specifically OpenAI's GPT-3.5. As illustrated in Fig. 1, the process begins with inputting course specifications in PDF format, followed by text extraction. Extracted verbs are mapped to cognitive levels in Bloom's taxonomy, which guides prompt generation for ChatGPT-based analysis. The model then evaluates the alignment between CLOs and assessment and teaching methods and strategies, providing feedback and suggestions, which are compiled into structured outputs for review.

A. Data Collection

The dataset used in this study consisted of 56 annotated CLO instances extracted from course specifications of a Master's program in Artificial Intelligence. Course specifications were retrieved from publicly available sources on the websites of Saudi universities. Each entry includes structured metadata such as course title, CLO code and description, teaching strategy, assessment method, and model-generated feedback. Data ingestion was performed, where batch extraction of course content was automated. The dataset covered multiple CLOs per course and preserved contextual elements such as program title and level, ensuring validity across hierarchical program structures.

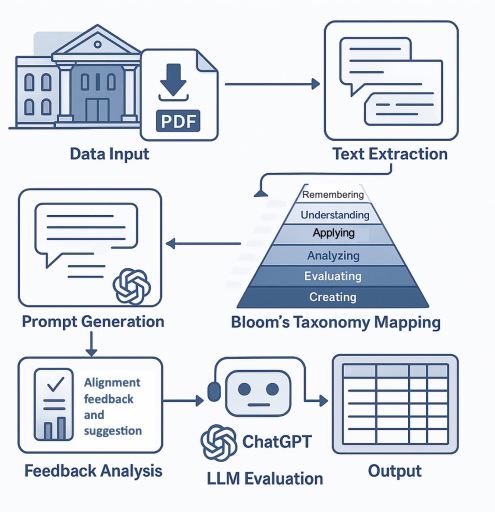


Fig. 1. Computational workflow for evaluating CLO-assessment alignment using Bloom's taxonomy and ChatGPT.

B. CLO Identification and Filtering

To ensure analytical rigor and consistency with national academic standards, only CLO entries that followed a standardized numerical format (e.g., 1.1, 2.2, 3.1) were included in the dataset. This structure directly reflects the official course specification template mandated by the National Center for Academic Accreditation and Evaluation (NCAAA), as specified in the official guidelines provided by the NCAAA [5]. Nearly all Saudi universities follow this core specification model as part of the broader transformation of higher education quality and accreditation in Saudi Arabia [17].

To avoid misinterpretation, the extraction process was designed to filter out non-actionable section headers such as "Knowledge," "Skills," or "Values," unless they were immediately followed by a clearly numbered sub-outcome. This rule-based filtering, implemented using regular expressions, minimized false positives and helped ensure that only assessable, action-oriented CLOs were analyzed.

For each valid CLO, the corresponding teaching strategy and assessment method were also extracted, in accordance with the tripartite structure outlined by the NCAAA. Assessment methods were categorized as either direct (e.g., quizzes, exams, case studies) or indirect (e.g., student surveys, course evaluations), enabling a detailed mapping between learning outcomes and how they are taught and assessed [5]. This structured alignment forms the backbone of outcome-based education and was critical to our study's analysis.

C. Cognitive Level Classification

To determine the cognitive demand embedded in each Course Learning Outcome (CLO), a verb-level classification algorithm was employed, grounded in Bloom's Revised Taxonomy [18]. This taxonomy remains a foundational framework in curriculum design and assessment, especially in outcome-based education models implemented by accreditation bodies such as the NCAAA [19]. The algorithm matched verbs within each CLO to a curated and expanded list of action verbs categorized across the six hierarchical levels of Bloom's cognitive domain:

- Remembering: list, define, recall
- Understanding: describe, explain, summarize
- Applying: implement, demonstrate, solve
- Analyzing: differentiate, compare, examine
- Evaluating: assess, critique, justify
- Creating: design, construct, develop

Each CLO text was scanned for one or more of these verbs, and the corresponding cognitive level was automatically assigned based on the match. This automated process was not only efficient but also allowed for objective categorization, ensuring consistency across all entries. Importantly, this form of classification provides a quantifiable lens through which CLOs can be evaluated for cognitive rigor, a key requirement in accredited program design.

Notably, approximately 86% of the CLOs analyzed contained clearly identifiable action verbs that could be confidently mapped to a Bloom's level. The remaining 14% were flagged as "unknown" due to abstract or vague phrasing, or due to formatting inconsistencies that hindered proper parsing (e.g., missing or malformed sentence structures). These entries were reviewed separately to avoid skewing the cognitive alignment analysis.

By integrating this classification step, the study enabled a structured comparison between intended learning outcomes and their respective assessment strategies. This alignment checking based on cognitive complexity, serves as a diagnostic tool for academic programs seeking to enhance the coherence and integrity of their curriculum.

D. AI-Assisted Evaluation and Hallucination Mitigation

Each CLO–assessment pair was evaluated using OpenAI's ChatGPT (GPT-3.5), which served as a reflective assistant rather than a final judge. To guide the model's reasoning and reduce the chances of irrelevant or inconsistent responses (commonly known as hallucinations), we designed a structured prompt template. This template included key elements such as:

- Program and course title
- CLO description
- Assessment method
- Teaching strategy
- The extracted Bloom's verb and its associated cognitive level

These components were chosen to provide the model with enough context to make informed and consistent evaluations. Recognizing the limitations of large language models, especially their tendency to fabricate or deviate from input, we implemented several best practices to keep the responses grounded and trustworthy.

First, we used prompt confinement, a strategy where prompts are written in a tightly scoped and consistent format.

Research has shown that this approach helps reduce factual drift and variability in model output [12], [20]. Second, we employed template-based reasoning, where the model was asked to reflect using logic similar to rubrics commonly used by educators [21]. Rather than asking the model to "know" the right answer, it was asked to reason based on the information provided.

Importantly, the model was not expected to draw on external or domain-specific knowledge. Instead, it was asked to evaluate only what was explicitly included in the prompt. After generation, responses were stored and reviewed for clarity, internal logic, and coherence. A portion of them were manually inspected to ensure that the feedback generated was interpretable and relevant.

It's worth emphasizing that the role of ChatGPT in this process is supportive, not authoritative. It serves as a first-pass evaluator, surfacing potential misalignments and offering justifications or improvement suggestions that can help curriculum designers reflect more deeply on the alignment between learning outcomes and assessment strategies. Human judgment remains central to the final decision-making process.

By combining structured prompting with thoughtful review, this approach enables scalable, replicable, and explainable evaluation of CLO–assessment alignment. It also shows promise for future integration into broader curriculum quality assurance systems suggesting the potential of large language models for educational auditing and curriculum evaluation workflows [22].

IV. Results

This section presents the results of a two-part evaluation of the AI-powered reflective tool designed to review the alignment between Course Learning Outcomes (CLOs), assessment methods, teaching strategies and cognitive complexity. The first part focuses on the tool's internal consistency, analyzing how well it identifies Bloom's taxonomy verbs, assigns cognitive levels, and interprets alignment across 56 CLOs. The second part examines how human experts perceive the tool's feedback. Two academic quality experts independently reviewed a sample of CLOs and rated the accuracy, usefulness, and cognitive alignment of the generated responses. Together, these findings offer a well-rounded view of how the tool performs in both automated checks and human-led evaluations. Section VI(A) explores patterns in Bloom's verbs, distributions of cognitive levels, and how frequently the tool flagged alignment issues. Section VI(B) highlights the level of agreement between experts and the AI tool, shedding light on where the model's evaluations aligned with expert judgment, and where improvements may be needed.

A. Self-Consistency Checks

To evaluate the internal consistency of ChatGPT's reflective evaluations across 56 Course Learning Outcomes (CLOs), a comprehensive analysis was performed on three key components: the Bloom's action verbs extracted from each CLO, the associated cognitive levels based on Bloom's Revised Taxonomy, and the nature of the corresponding assessment tasks. This triangulation aimed to determine whether the model's feedback logically corresponded with the expected learning complexity and pedagogical intent.

The CLOs were automatically parsed to detect the primary action verb and its associated Bloom's cognitive level. As illustrated in Fig. 2, the most commonly identified verbs included discuss, explain, analyze, and justify, spanning cognitive levels 2 (Understanding) to 5 (Evaluating). This distribution aligns with the higher-order thinking skills typically emphasized in postgraduate-level courses. A small number of CLOs were categorized under the "unknown" label, indicating either missing or ambiguous verbs that did not match the taxonomy database, highlighting potential issues in CLO phrasing or document formatting.

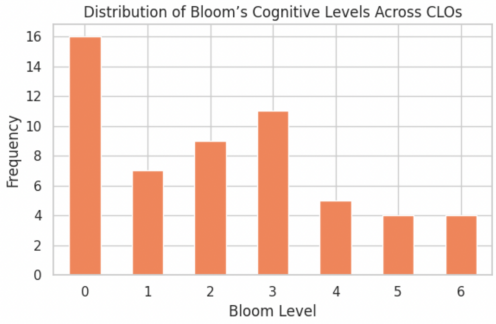


Fig. 2. Distribution of bloom's cognitive levels across CLOs.

Each ChatGPT-generated feedback response was then classified into one of three categories: Aligned, Misaligned, or Suggested. Feedback labeled as Aligned indicated that the model found no major inconsistencies between the CLO, its assessments, and teaching strategies. Misaligned denoted explicit concerns or mismatches raised by the model. Suggested included responses that, while not flagging critical misalignments, recommended improvements to strengthen alignment or clarity.

B. Illustrative Examples of Alignment and Misalignment

Table I presents two anonymized, representative CLO–assessment cases from our dataset, one *Aligned* and one *Misaligned*. For each case, we report the CLO text, Bloom level, assessment/teaching methods, the model's alignment label, and an abridged rationale highlighting specific alignment cues (e.g., verb–task cognitive match/mismatch, direct vs. indirect assessment, authenticity of performance tasks).

As shown in Fig. 3, a majority of responses were classified as Aligned, suggesting that many CLOs were appropriately constructed and assessed. However, a notable portion of feedback was marked as Misaligned or included improvement Suggestions, underscoring recurrent challenges in articulating learning outcomes and aligning them effectively with assessments and pedagogy.

These findings underscore the potential of structured LLM-based tools as formative quality assurance mechanisms. The distribution of Bloom's levels offers insight into instructional emphasis across the curriculum, while the alignment feedback highlights areas requiring further improvement in outcome

TABLE I. TWO COMPACT EXAMPLES: (*Left*) ALIGNED, (*Right*) MISALIGNED. TEXTS ARE ANONYMIZED AND LIGHTLY ABRIDGED

Field	Case A: Aligned	Case B: Misaligned
CLO	(example) "Describe core concepts of X and apply them to simple scenarios."	(example) "State definitions of X."
Bloom	Apply (L3)	Remember (L1)
Assess/Teach	Direct: exam, quiz, project; Indirect: survey / Lectures, guided exercises	Direct: quiz, exam; Indirect: survey / Lectures
Label	Aligned	Misaligned
Rationale	Tasks include a project and applied items that explicitly require L3 "apply", matching the CLO verb and expected level; a mix of direct/indirect evidence is present.	CLO targets L1 "remember" (state), yet assessment emphasizes higher-demand tasks (exam items requiring application/analysis) without corresponding CLOs; survey adds only indirect evidence, causing a verb–task mismatch.

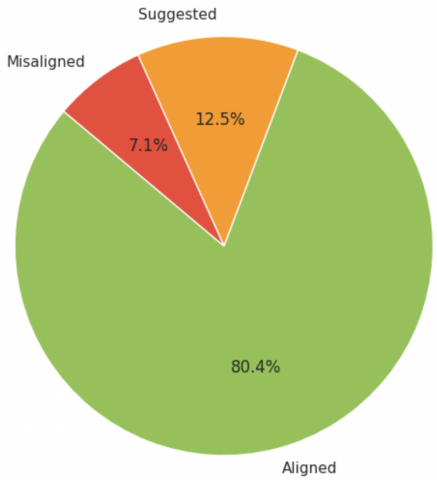


Fig. 3. Distribution of CLO–Assessment alignment outcomes.

construction and strategy selection. For curriculum designers and quality officers, these visualizations serve as practical dashboards, enabling them to detect trends, monitor inconsistencies, and prioritize reviews of misaligned CLOs. Crucially, this approach complements, rather than replaces, expert judgment by enhancing consistency, scalability, and transparency in curriculum evaluation workflows.

C. Human Expert Evaluation

To ensure the external validity of the AI-assisted evaluation and to provide a qualitative benchmark, a stratified sample of 12 Course Learning Outcomes (CLOs), representing approximately 20% of the full dataset, was independently reviewed by two academic quality experts. These experts were faculty members with substantial experience in course specification, learning outcomes design, and curriculum alignment in higher education.

Each expert received the original CLO statement, the associated assessment method, and the feedback generated by the AI-powered reflective tool (ChatGPT). They were asked to evaluate the AI's responses using a structured rubric built around three key dimensions, informed by best practices in AI-supported assessment [23], [24]:

- Accuracy: Whether the feedback correctly assessed the alignment between the CLO and the assessment.
- Usefulness: Whether the feedback provided practical value for improving CLO or course design.
- Cognitive Alignment: Whether the feedback appropriately matched the CLO's Bloom's level with its assessment method.

Ratings were recorded on a 5-point Likert scale (1 = Strongly Disagree, 5 = Strongly Agree), allowing for both descriptive and inter-rater agreement analysis.

Across the sample, the descriptive results showed strong overall agreement between the two experts, with most mean scores falling between 3 and 5. This suggests that the AI-generated feedback was generally perceived as accurate and useful. To statistically assess inter-rater consistency, Cohen's Kappa was calculated for each rubric dimension as shown in Table II. The results confirmed a moderate to substantial level of agreement, consistent with recent work highlighting the interpretability and limitations of large language models in educational tasks [25].

TABLE II. INTER-RATER AGREEMENT BASED ON COHEN'S KAPPA

Dimension	Kappa Score	Interpretation
Accuracy	0.41	Moderate agreement
Usefulness	0.12	Slight agreement
Cognitive Alignment	0.24	Fair agreement

This expert validation step reinforces the potential of LLMs to function as reflective aids in curriculum evaluation, provided that their outputs are critically reviewed by human experts, as part of a hybrid human–AI workflow.

Fig. 4 provides a side-by-side comparison of ratings from each expert across all CLOs and criteria. While the scores generally converged, certain items, particularly within the "Usefulness" dimension, showed divergence, highlighting the need for domain-specific refinement in feedback generation. The results show that the reflective tool provides reasonably consistent evaluations of CLO-alignment, particularly in identifying surface-level alignment mismatches. The tool's strength lies in its structured analysis based on Bloom's Taxonomy and its ability to assist with systematic curriculum review. However, lower inter-rater agreement on "Usefulness" suggests that some feedback lacked sufficient context or actionable depth. This emphasizes that while AI tools can support accreditation workflows, human oversight remains essential, especially when interpreting pedagogical intent or discipline-specific nuance. This expert validation underscores the tool's role as a decision-support system, offering scalable consistency while leaving room for human educational judgment.

V. DISCUSSION

The findings from this study underscore the promising role of structured, prompt-based large language models (LLMs)

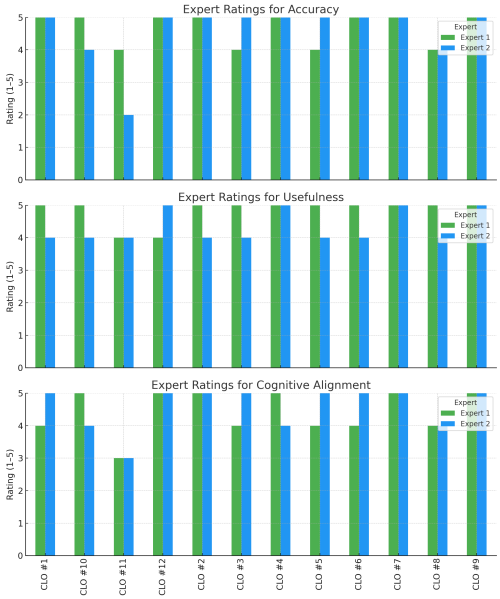


Fig. 4. Comparison of expert ratings for each CLO across three evaluation dimensions: accuracy, usefulness, and cognitive alignment.

in supporting curriculum evaluation tasks that require both cognitive interpretation and alignment with institutional standards. By focusing on Course Learning Outcomes (CLOs), their associated assessments, and pedagogical strategies, the reflective tool demonstrated an ability to surface misalignments and provide constructive, taxonomy-informed feedback.

A key insight emerging from the self-consistency analysis is the model's sensitivity to cognitive distinctions embedded in Bloom's Revised Taxonomy. The high prevalence of verbs corresponding to the "Understanding" through "Evaluating" levels aligns well with the nature of postgraduate course design, where deeper cognitive engagement is expected [26]. However, the relatively low frequency of "Creating" level verbs may suggest either an underrepresentation of higher-order synthesis tasks in course documentation or conservative formulation of CLOs that avoid explicitly committing to evaluative or generative assessments. This observation resonates with prior literature indicating that course developers often shy away from cognitive verbs associated with advanced competencies due to either misinterpretation or fear of overpromising learning deliverables [27], [13].

The model's classification of CLO-assessment relationships into "Aligned," "Misaligned," or "Suggested" reflects an operational advancement over prior implementations of LLMs in educational contexts, where feedback tends to be generic or contextually shallow [28], [29]. The structured prompt engineering and taxonomy-based scaffolding used in this study likely contributed to this improvement by narrowing the model's interpretive scope, a technique echoed in recent AI reliability research [12], [30]. These findings affirm that template-based confinement not only mitigates hallucinations but also enhances the model's ability to reason reflectively based on provided input rather than drawing from its latent knowledge alone.

Notably, the human expert evaluation results further validate the model's potential as a pedagogical aid. While moderate agreement on "Accuracy" and "Cognitive Alignment" suggests foundational reliability, the limited consensus on "Usefulness" points to an important limitation. Specifically, the model-generated feedback may lack domain-specific depth or contextual nuance, especially when CLOs involve interdisciplinary goals or implicit institutional constraints. This aligns with the findings of Luo et al. [31] and Iftene et al. [8], who caution that while LLMs can serve as reflective collaborators, they are not yet replacements for domain expertise. The divergences in expert ratings highlight the necessity of maintaining human oversight, particularly for evaluating the pedagogical appropriateness and practical feasibility of suggested improvements.

Importantly, the tool's value lies not in definitive judgment but in its capacity to structure and scale reflective evaluation. In contexts such as those governed by the NCAAA, where alignment and documentation rigor are essential, having a replicable first-pass reviewer that surfaces potential issues can reduce faculty workload and enable earlier intervention in curriculum development cycles. The use of visualizations (e.g., cognitive level distributions and alignment categorizations) enhances interpretability and provides a data-driven dashboard for institutional review, echoing similar recommendations in curriculum analytics literature [32], [33].

Nonetheless, the system is not without limitations. A small percentage of CLOs were unclassifiable due to ambiguous phrasing or formatting inconsistencies, highlighting ongoing challenges in automating quality assurance across diverse documentation standards. Furthermore, while the tool supports alignment evaluation, it does not currently assess whether the assessments themselves are valid measures of cognitive complexity, a task that likely requires deeper qualitative analysis or human triangulation.

VI. Conclusion

This study introduced a structured, prompt-based reflective framework that leverages ChatGPT to evaluate the alignment between Course Learning Outcomes (CLOs), assessment methods, and teaching strategies in line with NCAAA standards. Unlike general-purpose AI applications, our approach focused on targeted prompts grounded in Bloom's Taxonomy and the official course specification structure to generate consistent and relevant feedback.

Through self-consistency checks across 56 CLOs, the tool demonstrated a reasonable ability to detect Bloom's action verbs, classify cognitive levels, and flag potential misalignments in assessment alignment. A majority of the AI-generated feedback was classified as aligned, though a significant portion provided constructive suggestions or identified mismatches highlighting common design issues in course documentation. To validate the tool's reliability, two academic quality experts independently evaluated a stratified sample of CLOs. Their assessments showed moderate inter-rater agreement on accuracy, slight agreement on cognitive alignment, and limited agreement on usefulness suggesting that while the tool provides structured and technically sound evaluations, human oversight remains essential, particularly in interpreting nuanced instructional context.

Overall, this research demonstrates the potential of large language models to act as scalable and structured reflective tools in curriculum quality assurance. Rather than replacing educators, the system is designed to support them offering data-driven insights that can guide course improvement and strengthen institutional readiness for accreditation reviews.

DECLARATION

The author declares that she has no known competing financial interests or personal relationships that could have appeared to influence the work reported in this document.

REFERENCES

[1] J. Biggs and C. Tang, *Teaching for Quality Learning at University*, 4th ed. McGraw-Hill Education, 2011.

[2] O. Al-Diban, H. Alshareef, and M. Saleh, "Analyzing curriculum alignment gaps using bloom's taxonomy in saudi universities," *International Journal of Educational Research*, vol. 118, p. 102234, 2024.

[3] R. Mahmoud, A. Alshammari, and R. Almefleh, "Evaluating learning outcomes for accreditation: A case study in saudi higher education," *Education and Information Technologies*, vol. 29, no. 2, pp. 456–478, 2024.

[4] L. Ha, Y. Chen, and J. Rios, "Reflective course design with llms: Minimizing hallucination and enhancing outcome alignment," *Computers & Education: Artificial Intelligence*, vol. 5, p. 100177, 2024.

[5] National Center for Academic Accreditation and Evaluation (NCAAA), *National Qualifications Framework and Course Specification Guidelines*, Education and Training Evaluation Commission (ETEC), Saudi Arabia, 2021, available at: https://etec.gov.sa/en/productsandservices/NCAAA/Pages/default.aspx.

[6] J. Johansson, D. Broman, and L. Plahn, "Field courses for dummies? to what extent can chatgpt design a higher education field course," *Higher Education*, vol. 87, pp. 1–17, 2024.

[7] V. Authors, "Cdio 2024 proceedings," in *Proceedings of the 20th International CDIO Conference*. CDIO, 2024, https://cdio.org/.

[8] A. Iftene, I. Ciocoiu, and L. Mihăescu, "Language models and higher education: The use of chatgpt in romanian academic settings," *Education Sciences*, vol. 12, no. 541, 2024.

[9] E. A. Van Dis, J. Bollen, and W. Zuidema, "What can chatgpt not do in education? evaluating its effectiveness in assessing educational learning outcomes," *Educational Technology Research and Development*, 2024.

[10] M. Sahlgren, D. Holmer, and F. Olsson, "Educational assessment with llms: Opportunities and risks," *Educational Technology & Society*, vol. 26, no. 2, pp. 22–35, 2023, preprint available from RTH. [Online]. Available: https://rth.se/publications/llms-assessment.pdf

[11] M. Sallam, "Chatgpt utility in healthcare education, research, and practice: Systematic review on the promising perspectives and valid concerns," *Healthcare*, vol. 11, no. 6, p. 887, 2023. [Online]. Available: https://www.mdpi.com/2227-9032/11/6/887

[12] L. Reynolds and K. McDonell, "Prompt programming for large language models: Beyond the few-shot paradigm," https://arxiv.org/abs/2102.07350, 2021.

[13] A. Lubbe, E. Marais, and D. Kruger, "Cultivating independent thinkers: The triad of artificial intelligence, bloom's taxonomy and critical thinking in assessment pedagogy," *Education and Information Technologies*, 2025, in Press.

[14] L. Luo, H. Yang, and X. Yu, "Human–ai collaboration in education: Principles and practices," *BMC Medical Education*, vol. 23, no. 1, p. 341, 2023.

[15] H. Kissane, A. Schilling, and P. Krauss, "Probing internal representations of multi-word verbs in large language models," *arXiv preprint arXiv:2502.04789*, 2025. [Online]. Available: https://arxiv.org/abs/2502.04789

[16] S. Nurmatova and M. Altun, "A comprehensive review of bloom's taxonomy integration to enhancing novice efl educators' pedagogical impact," *Arab World English Journal (AWEJ)*, vol. 14, no. 3, pp. 380–388, 2023. [Online]. Available: https://ssrn.com/abstract=4593934

[17] A. Alhazmi, "Higher education reform in saudi arabia: The journey toward vision 2030," *Journal of Education and Learning*, vol. 11, no. 1, pp. 100–110, 2022.

[18] L. W. Anderson and D. R. Krathwohl, *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives*. New York: Longman, 2001.

[19] A. Bakharia, "Can chatgpt teach us about academic writing? large language models as ai writing tutors," *Media Education Research Journal*, vol. 9, no. 1, pp. 1–14, 2023. [Online]. Available: https://doi.org/10.5920/mep.20004

[20] X. Zhou, J. Liu, M. Sun *et al.*, "A survey of hallucination in large language models," *arXiv preprint arXiv:2302.08091*, 2023. [Online]. Available: https://arxiv.org/abs/2302.08091

[21] A. Lubbe, E. Marais, and D. Kruger, "Cultivating independent thinkers: The triad of artificial intelligence, bloom's taxonomy, and critical thinking in assessment pedagogy," *Education and Information Technologies*, vol. 29, no. 1, pp. 1–20, 2024.

[22] E. Kasneci, K. Sessler, H. Krosse, and G. Kasneci, "Chatgpt for good? on opportunities and challenges of large language models for education," *Learning and Individual Differences*, vol. 103, p. 102274, 2023.

[23] R. AlAli and Y. Wardat, "Opportunities and challenges of integrating generative artificial intelligence in education," *International Journal of Religion*, vol. 5, no. 7, pp. 784–793, 2024. [Online]. Available: https://doi.org/10.61707/8y29gv34

[24] O. Zawacki-Richter, V. I. Marín, M. Bond, and F. Gouverneur, "A systematic review of research on artificial intelligence applications in higher education: Learning, teaching and administration," *Educational Technology Research and Development*, vol. 69, no. 2, pp. 733–762, 2021.

[25] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx *et al.*, "On the opportunities and risks of foundation models," *arXiv preprint arXiv:2201.06935*, 2022. [Online]. Available: https://arxiv.org/abs/2201.06935

[26] Y. Luo, T. Liu, P. C.-I. Pang, D. McKay, Z. Chen, G. Buchanan, and S. Chang, "Enhanced bloom's educational taxonomy for fostering information literacy in the era of large language models," *arXiv preprint arXiv:2503.19434*, 2025. [Online]. Available: https://arxiv.org/abs/2503.19434

[27] J. Biggs and C. Tang, *Teaching for Quality Learning at University*, 4th ed. McGraw-Hill Education, 2011.

[28] E. A. Van Dis, J. Bollen, and W. Zuidema, "What can chatgpt not do in education? evaluating its effectiveness in assessing educational learning outcomes," *Educational Technology Research and Development*, 2024.

[29] M. Sahlgren, D. Holmer, and F. Olsson, "Educational assessment with llms: Opportunities and risks," *Educational Technology & Society*, vol. 26, no. 2, pp. 22–35, 2023.

[30] X. Zhou, J. Liu, M. Sun *et al.*, "A survey of hallucination in large language models," *arXiv preprint arXiv:2302.08091*, 2023.

[31] L. Luo, H. Yang, and X. Yu, "Human–ai collaboration in education: Principles and practices," *BMC Medical Education*, vol. 23, no. 1, p. 341, 2023.

[32] O. Al-Diban, H. Alshareef, and M. Saleh, "Analyzing curriculum alignment gaps using bloom's taxonomy in saudi universities," *International Journal of Educational Research*, vol. 118, p. 102234, 2024.

[33] R. Mahmoud, A. Alshammari, and R. Almefleh, "Evaluating learning outcomes for accreditation: A case study in saudi higher education," *Education and Information Technologies*, vol. 29, no. 2, pp. 456–478, 2024.

AI and 5G Integration for Smart City Energy Systems: A Systematic Literature Review

TALBI Chaymae[1], Rahmouni M'hamed[2], OUAHBI Younesse[3], ZITI Soumia[4]
Research Team of Intelligent Processing and Security of Systems (IPSS),
Department of Computer Science, Faculty of Sciences,
Mohammed V University in Rabat, Rabat, Morocco[1,3,4]
Research Team of Intelligent Processing and Security of Systems (IPSS),
Department of Computer Science, Ecole Normale Supérieure (ENS)
Mohammed V University in Rabat, Rabat, Morocco[2]

***Abstract*—Smart cities increasingly rely on Artificial Intelligence (AI), 5G, and Internet of Things (IoT) technologies to enhance energy management and support real-time decision-making in smart grids. This study presents a systematic literature review of recent research on the integration of AI and 5G in urban energy systems, with a focus on sustainability goals. It examines how these technologies are used for renewable energy integration, demand-side control, and predictive maintenance across smart environments. Using data from OpenAlex, Scopus, and Web of Science covering the period 2018 to 2025, the review was filtered by language, domain, and scientific relevance. Key findings reveal the use of machine learning models for forecasting, anomaly detection, and system optimization. The review also identifies technical, ethical, and infrastructural challenges, including data heterogeneity, limited interoperability, and regional inequalities in deployment. While AI and 5G offer promising capabilities for real-time monitoring and system automation, the literature shows persistent gaps in algorithm robustness and standardized integration frameworks. The paper emphasizes the need for validated, scalable solutions to achieve long-term energy sustainability. This review provides a clear overview of current trends and future directions in smart energy systems, contributing to a better understanding of how digital technologies shape the future of sustainable urban infrastructures.**

***Keywords*—Smart cities; artificial intelligence; 5G; energy management; smart grids; renewable energy integration; IoT; machine learning; sustainability**

I. Introduction

The rapid growth of the global urban population has brought urgent challenges in energy management, infrastructure stability, and operational flexibility. In response, the concept of smart cities has emerged, integrating advanced digital technologies to improve the efficiency and quality of urban services [2], [11]. Among the key enablers of this transformation, artificial intelligence (AI) and fifth-generation (5G) communication networks play a central role in reshaping energy systems to support sustainable urban development [9], [30].

Smart energy systems in urban environments depend on smart grids, intelligent buildings, IoT-connected devices, and distributed renewable energy sources [3], [14]. These systems require real-time monitoring, adaptive control, and future-oriented optimization capabilities increasingly supported by AI for prediction, anomaly detection, and intelligent energy flow management [19], [21], [24]. In parallel, 5G offers ultra-low latency and high-speed connectivity, enabling the instantaneous transmission of data across edge devices and cloud platforms, thereby supporting large-scale deployment of IoT-based control systems [23], [26], [31].

Despite their potential, the integration of AI and 5G in energy systems remains limited by technical and ethical barriers. Challenges include system fragmentation, lack of standardized protocols, scalability of algorithms, data heterogeneity, cybersecurity concerns, and uneven access to infrastructure across regions [9], [17], [30], [33], [34]. These gaps hinder the development of comprehensive frameworks capable of addressing energy performance, social equity, and long-term sustainability.

This study addresses these concerns by conducting a systematic literature review of publications indexed in OpenAlex, Scopus, and Web of Science, covering the period from 2018 to 2025. The objective is to assess how AI and 5G are integrated into smart city energy systems, identify their contributions and limitations, and explore future directions. The review is structured around the following research questions:

RQ1: How are AI and 5G used in smart city energy systems?

RQ2: What technical and ethical issues limit their integration?

RQ3: How is their impact on energy performance evaluated?

RQ4: What are the benefits and outcomes of AI-enhanced energy management in smart cities?

By addressing these questions, the study offers a structured understanding of the evolving role of AI and 5G in smart energy systems, highlighting both current contributions and the areas requiring further innovation and policy attention.

II. Methodology

During this study, we follow a systematic and data -driven method to ensure accurately and relevance in our findings. Fig. 1 emphasizes the five main steps in our research process. The first phase defines the scope of the study,with a focus on how artificial intelligence and 5G technologies contribute to the energy dimension of smart cities. In phase 2, we collect

literature from three large scientific platforms - Openlex, Web of Science and Scopus - advanced filtration strategies.

For OpenAlex, we processed the search using a combination of technical keywords such as “Energy” and “Smart”, and limited the results to open access tasks published between 2018 and 2025. We did not exclude - related domains such as drugs, psychology, chemistry and humanities to focus specially related to engineering. We also limited the selection from articles and book chapters related to IoT, Edge/Fog Computing, Smart System and AI-based urban technologies, and banned the language in English.

Step 3 includes inclusion criteria based on quality, relevance to energy domains and screening of extracted studies using AI and clear links for 5G integration. In Step 4, we use a thematic classification and bibliometric mapping to detect recurrent applications, technical barriers and regional contexts. Finally, phase 5 highlights the most important research contribution to applying AI and 5G to smart energy infrastructure in urban references, highlighting existing intervals and future instructions.

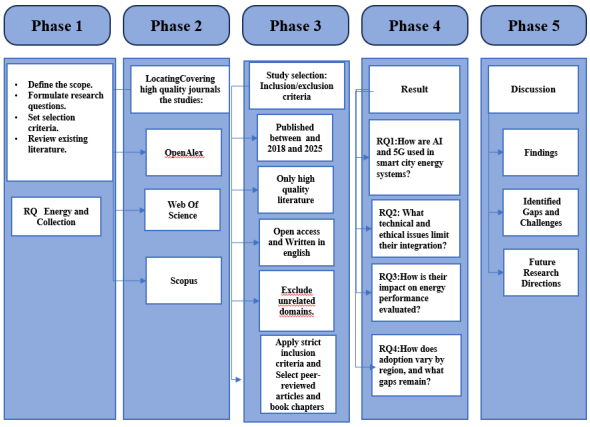


Fig. 1. Research process.

To continue the recovered literature base, we organized a descriptive book list of the final dataset. As shown in Fig. 2, the literature extends from 2018 to 2025 and includes a total of 1,350 documents from 570 different places of publication. These documents were written by 4,413 contributors, with an average of 3.62 co-author per paper, indicating a high level of cooperation research in the region. In particular, 33.78 percent of publications include international co-authors, reflecting global relevance and multi-creked interest in AI and 5G-based energy systems. The dataset contains 203 single author tasks, while the total number of references cited in all documents reaches 31,511, and emphasizes the depth of scholars and the prosperity of analyzed corpus. The average document age is 3.39 years, suggesting that the corpus is reflected in order to develop both recent and developed research mobility. In addition, the average quotation time per document is 16.22, which confirms the important education effect. Finally, we identified 546 separate author keywords, which act as a semantic foundation for thematic grouping and trend analysis in the later stages of this study.

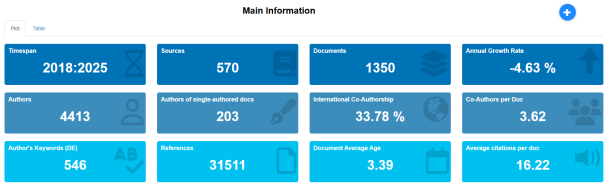


Fig. 2. Main information.

III. Results

Smart cities embody a modern strategy for urban development that leverages information and communication technologies (ICT) to make Urban more comfortable, durable and effective [1] [2]. At their core, smart cities aim to ensure a clean, cost-effective, and secure living environment for the citizens to live, work and play by organizing different systems, including energy, water, transport, public health and safety as showing in Fig. 3 [3].

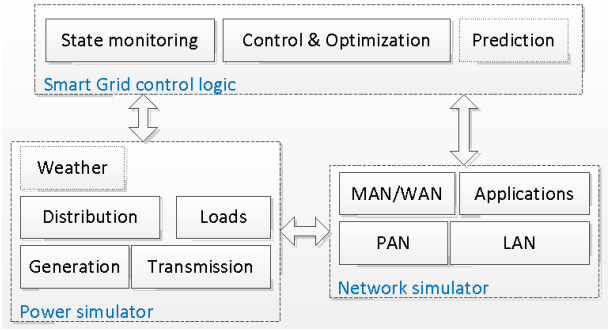


Fig. 3. Overview of a standard smart grid co-simulation environment.

The infrastructure of smart cities relies on a combination of interconnected technologies and subsystems. This includes smart grids, intelligent mobility solutions, adaptive traffic control, automated waste management, smart infrastructures, and systems for environmental monitoring [4]. The Internet of Things (IoT) plays an essential part by supporting real-time data collection and intelligent context-driven processing via interconnected sensors and actuators [1] [5].

The idea of smart urban control has developed with the emergence of recent technology which includes IoT, statistics analytics, and artificial intelligence [6] [7] [8]. These technology enable the optimization of strength intake, integration of renewable energy sources, discount of greenhouse gasoline emissions, and improvement in universal best of urban life [9][10] as showing in Fig. 4.

Smart city applications integrate artificial intelligence, data analytics, and advanced learning algorithms to enhance urban living standards for large populations [11] [12] The present and anticipated development of smart cities is marked by the convergence of various technologies, including 5G and blockchain infrastructure, which are expected to strengthen decision-making capabilities, anticipate urban dynamics, and support advanced IoT functionalities [13].

Smart city platforms support the collection and unified analysis of data originating from heterogeneous sources, in-

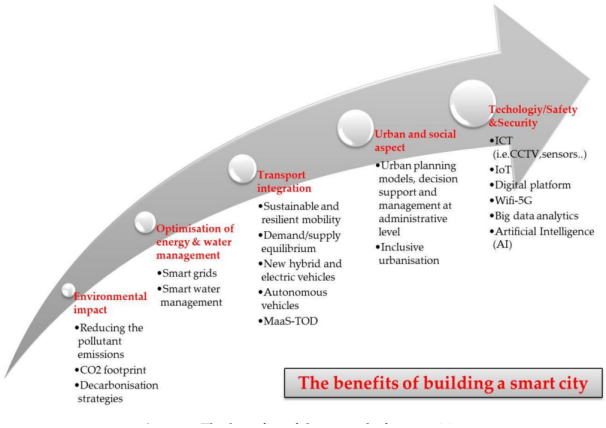


Fig. 4. The benefits of the spread of smart cities.

cluding systems for traffic observation, energy usage tracking, and environmental data sensing [14] [15]. This data -driven approach to city rule optimizes the operation and distribution of resources, and eventually leads to high energy efficiency and productivity.

Energy infrastructure plays an important role in smart cities, as all other urban works will eventually be closed if the energy is not available for a significant period [3]. Cities aim to build intelligent energy systems that support adaptive lighting, automated building systems, smart transport solutions, and energy distribution networks [16] as showing in Fig. 5.

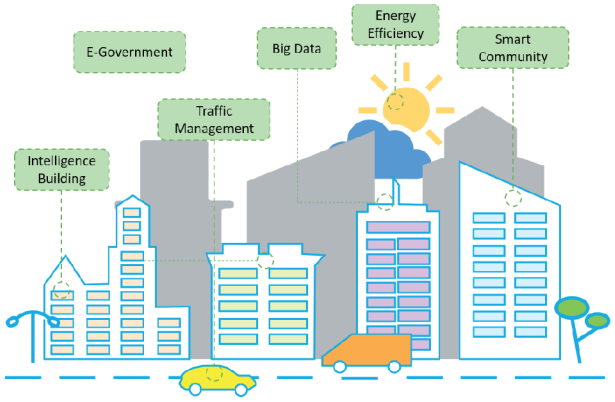


Fig. 5. Basic components of smart city.

Original research on smart cities focused on the principle, probe, design, development and implementation of smart, flexible and urban environments with low carbon. This included research on areas such as ICT and intelligent infrastructure, smart grid, energy -dense buildings, climate change mitigation, decision support systems and machine learning for machine learning for IoT, Big Data, AI and environmental and environmental [4].

A. Smart Cities and Energy Management

Energy management is among the most critical challenges of developing smart cities, which acts as the basis that all other urban functions depend on study [3]. The purpose of smart cities is to develop intelligent energy infrastructure that creates efficiency, increases economic growth and improves the quality of life through functions such as smart street lighting, intelligent buildings, smart mobility and power grid [16]. As part of the concept of "smart and sustainable cities," two flagship applications have emerged worldwide: smart electricity grids that promote renewable energy sources, and efficient mobility solutions that reduce automobile use and greenhouse gas emissions [17] A smart grid is one of the essential elements of smart city systems, alongside other intelligent infrastructure elements such as smart traffic lights, mobility solutions, buildings, waste management, and environmental monitoring [18].

Through self-healing designs, automation, and remote monitoring and control, smart energy grids modernize power systems. As a result of these systems, consumers will be able to make informed decisions and be able to safely integrate distributed renewable energy resources in order to improve overall energy efficiency [3]. In the context of future smart energy systems, algorithms are expected to infer user needs and consumption patterns by analyzing historical data, optimize and control the interconversion and distribution of energy vectors while taking engineering constraints and policy objectives into consideration [19].

Multi-energy vectors such as electric power, heat, and gas in urban environments enable integration of energy subsystems and support real-time operational control [19]. As embedded algorithms are increasingly deployed across multiple layers of the energy network, ranging from smart meters to real-time monitoring units, machine learning and intelligent computing tools are gaining relevance in supporting autonomous decision processes and adapting to evolving system behavior [19].

Modern smart city energy frameworks uses cutting-edge technology and data analytics to align and manage multiple urban infrastructures with the aim of increasing energy performance and operational output [14]. Smart grids are a key initiative to control energy consumption and optimize energy distribution, and integrate renewable energy technologies such as photovoltaic systems and wind energy converters within urban systems [14]. In smart cities, advanced smart grid frameworks are under development to address energy management challenges, including transactions and autonomous vehicle charging. [20]. These frameworks make use of a variety of technologies to enhance energy efficiency [20] as showing in Fig. 6, including deep learning, linear optimization, semantic technologies, and simulation. To integrate Information and Communication Technologies into urban development, smart cities adopt innovations including technologies like IoT, AI, Geographic Information Systems, digital twin models, remote sensing systems, intelligent transport infrastructure, and advanced energy grids [9] By implementing these tools, urban life is improved by reducing greenhouse gas emissions, optimizing energy consumption, incorporating sustainable energy inputs and lowering greenhouse gas emissions [9][10].

B. AI and Machine Learning Applications in Smart City Energy Systems

Artificial intelligence (AI) and machine learning (ML) are becoming increasingly vital for automated decision-making and responding to changes in energy system dynamics as embedded algorithms become more pervasive throughout energy

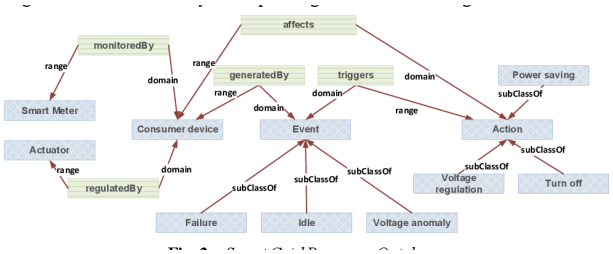


Fig. 6. Smart grid response ontology.

networks [19]. The concept of future smart energy systems includes autonomous algorithms capable of extracting user demand patterns and consumption behaviors from past data records, manage and coordinating the conversion and distribution processes of multiple energy carriers, while at the same time considering engineering constraints and policy objectives [19].

In smart city energy management, AI and ML applications are being leveraged for multiple purposes. Data-driven smart grid frameworks employ deep learning, linear optimization, semantic technology, and simulation to address complex energy management challenges including energy trading and autonomous vehicle charging [20]. As a result of these technologies, billions of people in urban areas are able to live more sustainably and improve their quality of life [11].

Optimization of renewable energy sources is one of AI's most significant contributions, which reduces carbon emissions and improves energy efficiency [21] [22]. With artificial intelligence and 5G technologies, energy providers are able to predict renewable energy production, such as when the sun will shine or the wind will blow, and adjust energy supply accordingly [23].

AI-driven smart grids enhance operational efficiency by dynamically adjusting to variations in energy supply and demand by leveraging real-time data analysis and adaptive learning algorithms, ensuring more stable and efficient distribution networks [21] [24] A smart building using AI-powered systems to forecast energy needs, improve the management of heating and cooling operations, and enhance the integration of renewable energy sources [21].

Several reviewed studies assess the energy-related impact of AI and 5G integration using quantitative indicators such as energy savings, carbon emission reductions, load balancing efficiency, and response time to peak demand. Evaluation methods include simulation-based energy modeling, real-time sensor data analysis, and predictive accuracy of AI-based demand forecasting systems. These metrics provide objective evidence of improved system performance and support the deployment of intelligent energy infrastructure in urban environments.

Besides energy management, machine learning, deep reinforcement learning, and artificial intelligence are being used in smart cities to develop intelligent transportation systems, cybersecurity, and 5G networking services. [25]. In transportation, AI and 5G can help reduce traffic congestion and emissions by optimizing traffic flow and identifying the most efficient vehicle routes, thereby decreasing time spent on the road and saving energy [23].

Researchers are addressing energy management challenges through a combination of IoT, 5G, and cloud computing, along with machine learning and deep learning approaches [26] [27]. Currently, researchers are working on developing algorithms for autonomously analyzing, predicting, and managing energy consumption in smart buildings and cities [28].

As artificial intelligence technology continues to develop, smart cities will apply AI more widely across intelligent transportation systems, energy management, and public safety to achieve more efficient urban management and services [29]. Despite the challenges regarding ethics and data management, AI is having a multi-dimensional impact on smart cities, affecting not only energy management, but also transportation and citizen engagement[30].

The evolution of smart cities has been supported by fundamental and applied research focused on AI and ML applications for environmental sustainability and climate change [4]. This research has laid the groundwork for current and future applications of these technologies in optimizing urban energy systems.

C. 5G Technology's Role in Smart City Energy Management

Smart city energy management systems depend on 5G wireless networks, which form the backbone of advanced energy systems.With 5G, IoT devices can communicate seamlessly and instantly, enabling real-time monitoring and control of energy assets across urban areas due to its ultra-low latency and high-speed connectivity [31]. As a result of enhanced connectivity, energy systems are more efficient and sustainable, particularly in applications such as smart grids, smart buildings, and intelligent transportation systems [31].

The integration of 5G with IoT technology creates powerful synergies for energy management in smart cities. Combining these technologies facilitates fast data transmission between communication channels and cloud storage, enabling efficient data retrieval and analysis for energy optimization [26] [27]. In renewable energy applications, 5G's instantaneous communication capabilities allow energy providers to quickly adjust their supply in response to AI predictions about solar and wind availability [23] as showing in Fig 7.

In addition to enhancing energy generation and distribution, 5G technology enhances transportation efficiency within smart cities, which impacts energy consumption overall. By optimizing traffic flow and identifying the most efficient routes for vehicles, 5G-enabled systems reduce congestion, reduce traffic time, and decrease emissions [23]. This demonstrates how 5G's benefits extend beyond direct energy management to address broader energy consumption patterns in urban environments.

The deployment of 5G networks promises to further enhance connectivity and enable more sophisticated IoT applications throughout smart cities [13]. As part of a wide spectrum of information technologies being applied to improve urban "smartness," 5G works alongside artificial intelligence, big data analytics, cloud computing, and IoT to create more responsive and efficient energy systems [32]. The continued evolution of these communication networks will play a pivotal role

in the future development of smart city energy management, providing the connectivity required for increasingly complex and integrated urban systems.

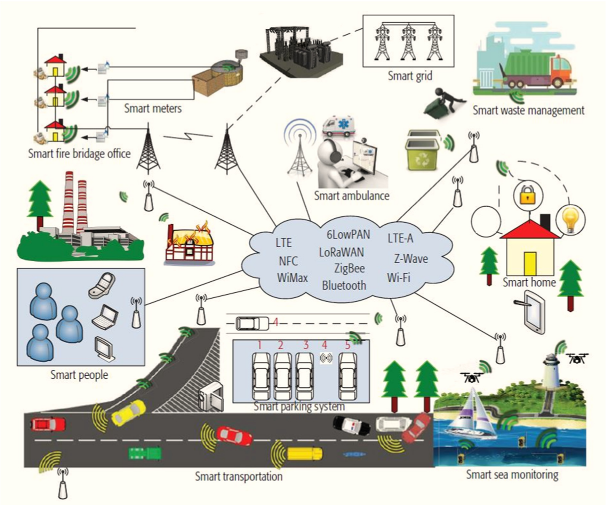


Fig. 7. Available communication technologies in various applications for smart cities.

D. Validation and Comparative Analysis with Existing Reviews

To enhance the scientific robustness of our study, we conducted a comparative validation by analyzing how our findings align with or extend those of recent systematic reviews addressing AI, 5G, and smart city energy systems. This approach confirms the coherence of our results while identifying the distinct contribution of our work.

Several prior studies have addressed parts of this research landscape. Bibri presented an integrated view of AI, IoT, and Big Data in sustainable smart cities, emphasizing technological convergence but overlooking infrastructure scalability and regional inequalities. O'Dwyer's review explored the role of AI in energy systems aligned with sustainability goals but did not incorporate 5G technologies or joint AI-5G integration frameworks. Santos examined the ethical and technical challenges of AI in smart cities, yet lacked consideration of telecommunication infrastructures essential for real-time operations. Camacho focused on AI-driven energy management but provided limited insights into system interoperability and IoT-5G data integration.

In contrast, our review takes a dual-focus approach that simultaneously addresses AI and 5G within smart energy systems. We offer a broader perspective that includes:

A combined analysis of technical and ethical challenges,

Regional disparities in deployment and access,

The need for real-time system responsiveness, algorithm robustness, and cross-platform interoperability.

Table I presents a structured synthesis of this comparative analysis. It clearly situates our contribution within the broader research context and highlights specific gaps in prior work that our study addresses.

IV. Discussion

A. Fragmentation Between Technological Layers

One key observation from the literature is the persistent disconnect between the development of artificial intelligence (AI) models and the implementation of 5G communication systems within smart city energy initiatives. Most studies concentrate on one isolated component. Some optimize machine learning algorithms for forecasting energy consumption or production. Others evaluate the performance of communication infrastructures, focusing on throughput, latency, or coverage.

Very few works integrate these components into a coherent, real-time energy management architecture. This siloed approach results in technical inconsistencies. For example, AI models are often designed without considering the limitations of underlying network infrastructures, which may not support the real-time data flows required. Conversely, 5G networks are sometimes deployed without embedding intelligent processing capabilities, limiting their contribution to autonomous decision-making.

This lack of integration across technological layers hinders the deployment of closed-loop, adaptive energy systems. Without synergy between AI and 5G, smart city platforms struggle to deliver timely, efficient, and scalable energy management in real-world environments.

B. Limited System-Level Evaluations

A recurrent issue across the reviewed studies is the limited presence of system-level validation. Most contributions focus on the performance of algorithms using isolated datasets or simulations conducted under fixed, controlled conditions. While these works are often rigorous in their methodological design, they neglect the complexities of real-world deployment.

There is little effort to assess the full data pipeline—from sensor-level data acquisition to transmission over 5G networks, edge processing, and final decision-making through AI models. The absence of integrated benchmarks reflecting interactions among these components limits the evaluation of key operational metrics such as scalability, resilience, and fault tolerance.

Without such validations, it becomes difficult to determine whether proposed solutions can maintain stability and responsiveness under realistic urban energy demands. This gap weakens the practical impact of many research efforts and hinders progress toward deployable, intelligent energy systems.

C. Underexplored Edge Intelligence Scenarios

Edge computing is widely acknowledged as a critical component for enabling distributed intelligence in smart energy systems. However, few studies propose concrete system architectures that effectively leverage edge nodes to reduce latency and optimize energy usage. The majority of contributions remain anchored in cloud-centric models, which can lead to communication overheads and increase the risk of delays in time-sensitive operations.

Despite its theoretical advantages, the integration of edge AI in energy-related use cases remains poorly formalized. The literature lacks structured frameworks for distributing AI

TABLE I. COMPARATIVE SUMMARY OF SELECTED LITERATURE REVIEWS

Author	cm**Scope of Review**	**Technology Focus**	**Main Gaps Identified**	**Limitations**
Bibri	cmSmart cities and sustainability	AI, IoT, Big Data	Ethics, convergence	No focus on regional disparities or scalability
O'Dwyer	cmSmart energy systems and sustainability	AI	Infrastructure, policies	No mention of 5G or real-time integration
Santos	cmAI challenges in smart sustainable cities	AI	Technical and ethical	Does not link with communication infrastructure
Camacho	cmAI for energy management in smart cities	AI	Algorithm limitations	Limited analysis of IoT/5G integration
Our review	cmAI and 5G in smart city energy systems	AI + 5G + IoT	Scalability, ethics, region	Integrated and comparative approach

tasks between edge and core nodes. In particular, there is limited work on adaptive task allocation strategies that consider dynamic factors such as local energy demand, network congestion, and the uncertainty of AI predictions.

This shortfall limits the development of responsive, low-latency energy systems capable of autonomous decision-making close to the data source.

D. Geographic and Contextual Imbalance

The geographic scope of the reviewed studies reveals a strong concentration in high-income countries with advanced technological infrastructures. These contexts often benefit from stable connectivity, standardized hardware, and supportive regulatory environments. In contrast, urban energy systems in low and middle income regions where networks are less reliable, devices are more heterogeneous, and policies are often fragmented remain largely absent from the literature.

This imbalance reduces the external validity of current research. Solutions optimized for well-resourced environments may not scale or adapt to settings with infrastructural and institutional constraints. To enhance applicability, future work must incorporate context-sensitive approaches that account for disparities in connectivity, device interoperability, governance models, and energy usage profiles.

E. Methodological Diversity and Lack of Convergence

The literature demonstrates a broad methodological range, encompassing supervised learning, reinforcement learning, federated approaches, and hybrid architectures. However, this diversity is not accompanied by standardized evaluation metrics or deployment protocols. The lack of convergence in assessment criteria makes cross-study comparisons unreliable and limits the identification of effective design patterns.

This fragmentation also poses challenges for reproducibility. Without shared benchmarks or experimental frameworks, replicating and validating results becomes complex, slowing down methodological refinement and large-scale adoption.

Advancing the field requires the adoption of common evaluation protocols, multi-criteria performance assessments, and the availability of open-access datasets to support fair benchmarking and collaborative development.

F. Gaps and Challenges in Smart City Energy Systems

Despite the promise of smart city energy systems, several critical gaps and challenges remain to be addressed:

Interoperability and Complexity of the System: In smart cities, interconnected tools, platforms, and sensors pose a challenge, but interoperability must be ensured [9]. Integration of various technological components requires standardized protocols, which are not yet fully established across all systems.

Management and Analysis of Data: Smart city infrastructure generates a great deal of data that presents significant challenges for gathering and analyzing [9]. Due to its volume, velocity, and variety, advanced processing capabilities are needed beyond what is currently available. Security and Privacy Concerns: As smart energy systems collect more and more data, privacy and security concerns must be addressed [9].

Algorithm Limitations: Existing studies often fail to address the full complexity and uncertainty of real-world scenarios in energy optimization, creating a gap between theoretical models and practical implementation [33].

Implementation of Advanced Algorithms: The energy consumption of smart buildings and cities still requires more sophisticated algorithms capable of automatically analyzing, predicting, and managing it [28].

AI Implementation Gaps: Technological gaps exist specifically when implementing artificial intelligence in smart cities, affecting the potential optimization of energy systems [34].

System Flexibility and Resilience: Renewable energy sources are becoming more prevalent, and modern energy management solutions do not offer the flexibility and resilience required to adapt to changing energy landscapes [28].

Integration of Renewable Energy: The integration of renewable energy sources into existing infrastructure remains a significant challenge for smart cities despite their desire to integrate renewable energy sources [9].

Comprehensive Energy Optimization: Despite advances in evolutionary algorithms and deep reinforcement learning, there is a need for more comprehensive and robust solutions for smart city energy operations that can handle real-world complexities [33].

Coordination Across Urban Systems: Achieving the holistic approach particular to smart cities requires coordinated and integrated social, environmental, and economic interventions across multiple sectors, presenting significant coordination challenges [9] [10].

G. Sustainability and Long-Term Perspectives

Sustainability concerns remain largely overlooked in the reviewed literature. Key aspects such as the energy consumption of AI models, the carbon footprint of 5G infrastructure, and the lifecycle impact of deployed systems are rarely quantified or

integrated into system design. This omission is critical, given that smart cities are expected to reduce emissions and promote energy efficiency.

There is a contradiction between these objectives and the current lack of energy-aware intelligence. Most AI-5G solutions are evaluated in terms of performance, with minimal attention to their environmental cost.

Future work should prioritize the development of resource-efficient AI models and establish methodologies for assessing the ecological impact of intelligent infrastructures. Embedding sustainability into design choices is essential to ensure long-term viability and alignment with climate goals.

V. Conclusion

This review explored the integration of artificial intelligence and 5G technologies into smart city energy systems through a systematic analysis of recent literature. The study identified their core applications in predictive analytics, load optimization, demand forecasting, and energy automation. AI contributes to real-time decision-making, while 5G ensures fast, reliable communication for distributed systems. Technical and ethical challenges remain, including algorithm transparency, data quality, privacy concerns, and unequal access to infrastructure. Evaluation methods found in the literature include simulation modeling, energy performance indicators, and predictive accuracy metrics.

Despite these advancements, several gaps persist regarding interoperability, standardized frameworks, and large-scale deployment. Future research should address cross-platform integration, improve data-sharing protocols, and validate models with real-world urban data. There is also a need to explore AI and 5G synergies in dynamic contexts such as mobility-energy coupling and disaster-resilient infrastructures. This study provides a roadmap for researchers and urban planners seeking to build energy-efficient, responsive, and sustainable cities using intelligent technologies.

References

[1] Yu-Hsiu Lin and Yu-Chen Hu, "Residential Consumer-Centric Demand-Side Management Based on Energy Disaggregation-Piloting Constrained Swarm Intelligence: Towards Edge Computing," *Italian National Conference on Sensors*, 2018.

[2] A. Zanella, Nicola Bui, A. Castellani, Lorenzo Vangelista, Michele Zorzi, "Internet of Things for Smart Cities," *IEEE Internet of Things Journal*, 2014.

[3] E. Tsampasis, L. Sarakis, H. Leligou, T. Zahariadis, J. Garofalakis, "Novel Simulation Approaches for Smart Grids," *J. Sens. Actuator Networks*, 2016.

[4] Elias Simon Bibri, John Krogstie, Amin Kaboli, Alexandre Alahi, "Smarter eco-cities and their leading-edge artificial intelligence of things solutions for environmental sustainability: A comprehensive systematic review," *Environmental Science and Ecotechnology*, 2023.

[5] Chun-Feng Liao, Peng-Yu Chen, "ROSA: Resource-Oriented Service Management Schemes for Web of Things in a Smart Home," *Italian National Conference on Sensors*, 2017.

[6] Alaa Ababneh, "Smart urban management of green space," *Journal of Design for Resilience in Architecture and Planning*, 2023.

[7] Y. Khan, M. M. Su'ud, M. M. Alam, Syed Fayaz Ahmad, A. Y. B. Ahmad Ayassrah, Nasir D. Khan, "Application of Internet of Things (IoT) in Sustainable Supply Chain Management," *Sustainability*, 2022.

[8] Qi Liu, H. Ullah, W. Wan, Zhangyou Peng, Li Hou, T. Qu et al., "Analysis of Green Spaces by Utilizing Big Data to Support Smart Cities and Environment: A Case Study About the City Center of Shanghai," *ISPRS Int. J. Geo Inf.*, 2020.

[9] Saeed Esfandi, S. Tayebi, John Byrne, Job Taminiau, Golkou Giyahchi, Seyed Ali Alavi, "Smart Cities and Urban Energy Planning: An Advanced Review of Promises and Challenges," *Smart Cities*, 2024.

[10] T. Campisi, A. Severino, Muhammad Ahmad Al-Rashid, G. Pau, "The Development of the Smart Cities in the Connected and Autonomous Vehicles (CAVs) Era: From Mobility Patterns to Scaling in Cities," *Infrastructures*, 2021.

[11] Md. Shelim Miah, Ruhul Amin, "Role of Technology in the Development of Smart Cities," *Engineering International*, 2020.

[12] Praveen Kumar Donepudi, "Crossing Point of Artificial Intelligence in Cybersecurity," , 2021.

[13] Eko Nursanty, Astari Wulandari, Djudjun Rusmiatmoko, Wawan Destiawan, "Redefining Urban Authenticity in the Digital Era: A Smart City Perspective," *E3S Web of Conferences*, 2024.

[14] Rui Ge, Shan Xu, Mirzat Ullah, Peter Mark, "Assessing the impact of urban planning policies on renewable energy: A case a China using the DID estimation model," *Heliyon*, 2024.

[15] Zhang Jiang, Chenghao Xu, "Disrupting the Technology Innovation Efficiency of Manufacturing Enterprises Through Digital Technology Promotion: An Evidence of 5G Technology Construction in China," *IEEE Transactions on Engineering Management*, 2024.

[16] Hossny Azizalrahman, Valid Hasyimi, "Towards a Generic Framework for Smart Cities," *Smart Urban Development*, 2019.

[17] S. Bibri, Alahi Alexandre, Ayyoob Sharifi, J. Krogstie, "Environmentally sustainable smart cities and their converging AI, IoT, and big data technologies and solutions: an integrated approach to an extensive literature review," *Energy Informatics*, 2023.

[18] Abdellah Chehri, I. Fofana, Xiaoming Yang, "Security Risk Modeling in Smart Grid Critical Infrastructures in the Era of Big Data and Artificial Intelligence," *Sustainability*, 2021.

[19] E. O'Dwyer, Indranil Pan, S. Acha, N. Shah, "Smart energy systems for sustainable smart cities: Current developments, trends and future directions," *Applied Energy*, 2019.

[20] N. Petrovic, Djordje Kocić, "Data-driven framework for energy-efficient smart cities," *Serbian Journal of Electrical Engineering*, 2020.

[21] Erik Karger, Aristide Rothweiler, Tim Brée, Frederik Ahlemann, "Building the Smart City of Tomorrow: A Bibliometric Analysis of Artificial Intelligence in Urbanization," *Urban Science*, 2025.

[22] Yuyang Li, Lihua Niu, "Prospects of Smart Cities in Energy Saving and Transportation Data Visualization under the Background of Dual Carbon," *SHS Web of Conferences*, 2024.

[23] M. Alahi, Arsanchai Sukkuea, F. Tina, A. Nag, W. Kurdthongmee, K. Suwannarat et al., "Integration of IoT-Enabled Technologies and Artificial Intelligence (AI) for Smart City Scenario: Recent Advancements and Future Trends," *Italian National Conference on Sensors*, 2023.

[24] J. Jiao, "Application and prospect of artificial intelligence in smart grid," *IOP Conference Series: Earth and Environment*, 2020.

[25] Anggy Giri Prawiyogi, Suryari Purnama, Lista Meria, "Smart Cities Using Machine Learning and Intelligent Applications (ITALIC)," *International Transactions on Artificial Intelligence (ITALIC)*, 2022.

[26] Kunal Sharma, Dasarathy A K, Ramakant Upadhyay, Kuldeep Singh Kulhar, "Review of Renewable Energy, Sustainability Concerns, and Climate Solutions for Smart Cities," *E3S Web of Conferences*, 2024.

[27] M. Humayun, M. Alsaqer, Noor Zaman Jhanjhi, "Energy Optimization for Smart Cities Using IoT," *Applied Artificial Intelligence*, 2022.

[28] Maryam Nikpour, Parisa Behvand Yousefi, H. Jafarzadeh, Kasra Danesh, Roya Shomali, Saeed Asadi et al., "Intelligent Energy Management with IoT Framework in Smart Cities Using Intelligent Analysis: An Application of Machine Learning Methods for Complex Networks and Systems," , 2023.

[29] Jianqiu Wang, Dr. Anukul Tamprasirt, "Modular Floating City: The Road to Future Development of Smart Cities," *Journal of Civil Engineering and Urban Planning*, 2023.

[30] Simone Santos, Jéssyka Vilela, Thiago Carvalho, Thiago Rocha, Thales Candido, Vinícius Bezerra et al., "Artificial Intelligence in Sustainable Smart Cities: A Systematic Study on Applications, Benefits, Challenges, and Solutions," *International Conference on Enterprise Information Systems*, 2024.

[31] Priyank Mishra, G. Singh, "Energy Management Systems in Sustainable Smart Cities Based on the Internet of Energy: A Technical Review," *Energies*, 2023.

[32] Nelson Pacheco Rocha, A. Dias, G. Santinha, Mário Rodrigues, Carlos Rodrigues, A. Queirós et al., "Systematic literature review of context-awareness applications supported by smart cities' infrastructures," *SN Applied Sciences*, 2022.

[33] Fenghua Liu, Xiaoming Li, "Integrating AI Deep Reinforcement Learning With Evolutionary Algorithms for Advanced Threat Detection in Smart City Energy Management," *IEEE Access*, 2024.

[34] J. Camacho, Bernabé Aguirre, Pedro Ponce, Brian Anthony, Arturo Molina, "Leveraging Artificial Intelligence to Bolster the Energy Sector in Smart Cities: A Literature Review," *Energies*, 2024.

State-of-the-Art in Software Security Visualization: A Systematic Review

Ishara Devendra[1], Chaman Wijesiriwardana[2], Prasad Wimalaratne[3]
Department of Information Technology, University of Moratuwa, Sri Lanka[1,2]
Faculty of Graduate Studies, University of Colombo, Sri Lanka[1]
Department of Communication and Media Technologies,
University of Colombo School of Computing, Colombo 07, Sri Lanka[3]

***Abstract*—Software security visualization is an interdisciplinary field that combines the technical complexity of cybersecurity, including threat intelligence and compliance monitoring, with visual analytics, transforming complex security data into easily digestible visual formats. As software systems get more complex and the threat landscape evolves, traditional text-based and numerical methods for analyzing and interpreting security concerns become increasingly ineffective. The purpose of this paper is to systematically review existing research and create a comprehensive taxonomy of software security visualization techniques through literature, categorizing these techniques into four types: graph-based, notation-based, matrix-based, and metaphor-based visualization. This systematic review explores over 60 recent key research papers in software security visualization, highlighting its key issues, recent advancements, and prospective future research directions. From the comprehensive analysis, the two main areas were distinctly highlighted as extensive software development visualization, focusing on advanced methods for depicting software architecture: operational security visualization and cybersecurity visualization. The findings highlight the necessity for innovative visualization techniques that adapt to the evolving security landscape, with practical implications for enhancing threat detection, improving security response strategies, and guiding future research.**

Keywords*—Security visualization; vulnerability analysis; threat intelligence; compliance monitoring

I. Introduction

The increasing use of software has introduced significant security challenges, prompting the industry to develop best practices to protect applications that handle sensitive information. The critical need for robust software security is evident, as vulnerabilities can lead to severe consequences, including data breaches, unauthorized control, and service disruptions, thereby compromising security policies [1], [2], [3]. Although considerable research has been conducted to prevent these vulnerabilities and automate their detection, the effectiveness of these efforts remains limited due to the growing complexity of software systems [4], [5].

Software security visualization has emerged as a promising approach to address these challenges by providing intuitive and interactive visual representations of security data [6]. These techniques enhance the comprehension of security vulnerabilities, facilitate threat detection, and support the maintenance of secure codebases [7]. However, despite its potential, there is a lack of comprehensive studies that systematically analyze the effectiveness of different software security visualization techniques and identify the specific areas where these techniques can be most impactful [8].

This review aims to fill this gap by conducting a systematic analysis of existing research in the field of software security visualization. By examining key studies and identifying the strengths and weaknesses of current visualization techniques, this paper highlights the critical areas where further research is needed and suggests directions for future development.

The paper is organized as follows: Section II discusses the research methodologies, including the study selection criteria. Section III presents an analysis of the findings, organized into comprehensive software development visualization, operational security visualization, and cybersecurity visualization. Section IV explores how software security visualization techniques are applied in real-world contexts. Section V discusses the key findings, including the categorization of visualization methods and their impact. Section VI concludes by emphasizing the importance of visualization in addressing cybersecurity challenges and supporting effective threat analysis.

II. Methodology

This section outlines the systematic approach used to review relevant studies, ensuring a comprehensive and unbiased selection of sources. It outlines the process for identifying, evaluating, and selecting studies based on predefined criteria to ensure the reliability and validity of the review.

A. Search Strategy

To ensure a comprehensive literature review, a systematic search strategy was employed following the PRISMA guidelines [9]. Databases such as IEEE Xplore, ACM Digital Library, SpringerLink, and ScienceDirect were searched for articles published over the last five years (2019–2024). The search terms used included combinations of "software security visualization," "cybersecurity visualization tools," "security data visualization," and related keywords. The search strategy was designed to capture all relevant studies in the field of software security visualization [10], [11].

B. Eligibility Criteria

Inclusion criteria for studies were rigorously defined as follows:

- Published in peer-reviewed journals or conference proceedings [9].

- Focus on visualization techniques specifically used in software security [10].
- Provision of empirical results concerning the effectiveness of visualization tools in cybersecurity.

Studies were excluded if they met any of the following criteria:

- Non-English language publications.
- Opinion pieces, editorials, or non-empirical articles.
- Studies not directly related to the field of software security visualization [11], [12].

C. Study Selection

The initial search yielded a total of 120 records. A detailed multi-step selection process was then employed:

1) Duplicate removal: Automated and manual processes were used to identify and remove duplicate records, resulting in 90 unique studies [9].

2) Title and abstract screening: The titles and abstracts of these studies were independently reviewed by two researchers to assess relevance. This step eliminated studies that did not align with the predefined inclusion criteria, resulting in 70 studies for full-text review [10].

3) Full-text review: The remaining articles were subjected to a detailed full-text review to assess their adherence to the eligibility criteria. This involved critically assessing the study's methodology, focus, and contributions to the field. In cases where the relevance was unclear, a third reviewer was consulted to reach a consensus [13].

4) Final selection: After the full-text review, 61 studies were deemed suitable for inclusion in the systematic review. The entire selection process was meticulously documented, and a PRISMA flow diagram was created to represent the screening and inclusion process visually [14].

D. Data Extraction

Data were extracted from each study by two independent reviewers using a standardized form [9]. The extracted information included authors, year of publication, study objectives, visualization techniques used, main findings, and contributions to software security visualization. Any discrepancies between reviewers were resolved through discussion or consultation with a third reviewer [10].

E. Quality Assessment

The quality of the included studies was assessed using the Critical Appraisal Skills Programme (CASP) checklist. This assessment focused on the methodological rigor and credibility of the studies, evaluating aspects such as the clarity of data presentation, the appropriateness of the analytical methods, and the generalizability of the findings [15].

F. Synthesis of Results

The data were synthesized qualitatively, categorizing studies by their focus areas: comprehensive software development visualization, operational security visualization, cybersecurity visualization, and information security visualization. Trends, commonalities, and divergences across studies were analyzed to identify the state of the art and gaps in the current literature [14].

III. Overview of Software Security Visualization

A systematic literature review has demonstrated the state-of-the-art of software visualization, aiding developers in selecting suitable visualization tools.

This study observed that most software visualizations are displayed on computer screens, and several approaches that are based on the structure of the system are displayed in 3D environments with city metaphors [16]. However, despite the advancements in visualization techniques, gaps remain, particularly in the application of these tools across different domains and user levels.

A study has reviewed 54 publications on Cyber Situational Awareness (CSA) visualizations, emphasizing the need for more comprehensive visualizations that support all levels of CSA and cater to a broader range of stakeholders. They identified gaps in existing CSA visualizations, such as a lack of visualizations for managers and non-expert users, and a focus on threat information over impact information and response plans [17]. This gap highlights the need for more user-centric visualization tools that non-technical stakeholders can effectively utilize, a critical area that remains underexplored.

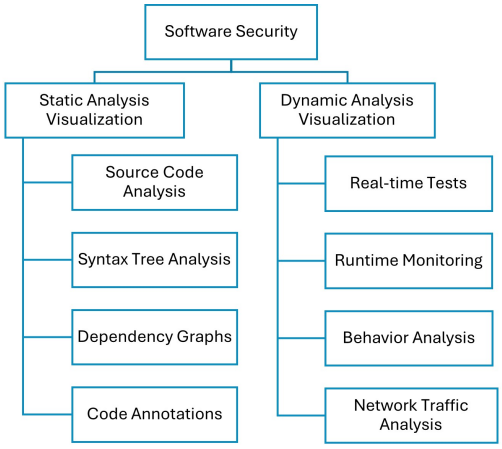


Fig. 1. Dynamic and static analysis visualization.

Another study has emphasized the need for standardized evaluation metrics for cybersecurity visualizations, highlighting methodological gaps and proposing potential research directions to establish a unified framework for evaluating the effectiveness of security visualizations [18]. Despite these proposals, a consensus on standardized metrics is still lacking, leading to inconsistencies in how visualization tools are evaluated and compared across different studies. Fig. 1 categorizes software security visualization into two main types: Static Analysis Visualization and Dynamic Analysis Visualization.

A. Static Analysis Visualization in Software Security

Static analysis visualization is crucial for understanding and identifying static code-based vulnerabilities without executing the program. This approach improves the efficiency and effectiveness of security assessments by providing clear visual perspectives into complex code structures and potential security issues.

Mollerstrom and Roos highlighted the critical role of threat analysis, particularly in making potential threats more understandable and actionable. It focuses on visualizing data flow within a program to assist in manual code reviews, making it easier to identify security vulnerabilities [19]. However, while visualization techniques are effective in static analysis, they cannot often detect runtime issues, limiting their applicability in dynamic environments.

Recent studies have further explored these methodologies in the context of container security, highlighting the use of static code analysis in identifying and mitigating vulnerabilities within software architectures [20]. Despite these advancements, the integration of static analysis with other security assessment tools remains a challenge, particularly in maintaining accuracy and scalability across different environments.

Similarly, Senanayake et al. reviewed methods for detecting vulnerabilities in Android source code, while emphasizing the role of visualization in presenting the identified vulnerabilities to the software practitioners [4]. However, the effectiveness of these visualization techniques in large-scale software systems with complex dependencies has not been thoroughly examined, highlighting a gap in the current research.

Recent studies have highlighted the ability to use deep learning methodologies with visualization techniques to increase the efficiency of vulnerability detection [21], [22]. While deep learning enhances the accuracy of detection, its integration with existing visualization tools poses challenges, particularly in terms of interpretability and the potential for false positives.

Studies by Fedorchenko et al. and Piantadosi et al. discussed models and approaches for exploit analysis and smart contract security, respectively, highlighting the importance of visual representations in mitigating and understanding security risk [23], [24]. These studies underscore the importance of context-aware visualizations that can adapt to different threat environments, a feature that is often missing in current tools.

B. Dynamic Analysis Visualization in Software Security

Dynamic Analysis Visualization plays a crucial role in identifying and understanding security vulnerabilities in real time when software executes. This approach enhances static code analysis by demonstrating runtime issues that might not be detected through static code analysis. Recent studies have highlighted the significance of understanding coding vulnerabilities among various languages and the utilization of vulnerability detection tools, respectively, both of which benefit from dynamic analysis in capturing real-time issues [25], [26]. However, these studies often lack a comparative analysis of how different tools detect complex, multi-stage attacks.

Alaoui and Nfaoui have noted that dynamic analysis enhanced with visualization can improve the detection of unknown threats, and the importance of using deep learning approaches in identifying web application attacks [22]. While this approach is promising, the challenges of integrating dynamic analysis with existing security frameworks and ensuring the scalability of these solutions are still underexplored. Similarly, Cerny et al. and Parker et al. demonstrate the importance of dynamic analysis to manage evolving system architecture and operational inefficiencies while discussing the reconstruction and visualization of microservice architecture and runtime antipattern identification, respectively [27], [28]. Some researchers have focused on detecting and visualizing complex vulnerabilities such as TOCTOU race conditions and multistage cyber-attacks, where dynamic analysis provides crucial insights into the temporal aspects of security threats [29]. However, the effectiveness of these techniques in real-world scenarios, particularly in large-scale distributed systems, remains underevaluated, representing another gap in the current research.

C. Software Security Visualization Techniques / Tools

Software security visualization techniques and tools play a critical role in enhancing an understanding of security vulnerabilities in software systems. Recent studies have highlighted the significant importance of visualization techniques in understanding software architecture and evolution, categorizing them into graph-based, notation-based, matrix-based, and metaphor-based types [27]. While these categorizations are helpful, they often overlook the potential overlaps between different types of visualizations, which could lead to more integrated and effective tools. Some studies have focused on the requirements and evaluation of software visualization tools, highlighting the need for scalability, interactivity, and rigorous validation methods to ensure their effectiveness [30]. However, the lack of standardized evaluation metrics remains a significant barrier to comparing and improving these tools, as noted in the literature.

A recent study has reviewed the importance of evaluating visualization tools to improve user performance and ensure reliable results. It has explored 3D visualization and software evolution techniques, highlighting their capacity to provide comprehensive views of software structures and changes over time [31], [32]. While 3D visualization offers significant advantages, its practical implementation in real-world software development environments is often limited by high computational costs and the steep learning curve required for effective use.

The visualization of code smells and software quality metrics was reviewed, emphasizing the role of identifying areas prone to security vulnerabilities [33]. Integrating these visualizations into continuous integration/continuous deployment (CI/CD) pipelines is still a challenge, limiting their usefulness in agile development environments. Since software analysts need to read documents and source codes and synthesize trace data from multiple resources, the visualization for security analysis is usually labor-intensive [34]. Recent studies highlighted that to resolve this problem, there is a requirement for automated analysis tools.

One such study proposed a threat-hunting architecture using a machine learning approach, which is a practical

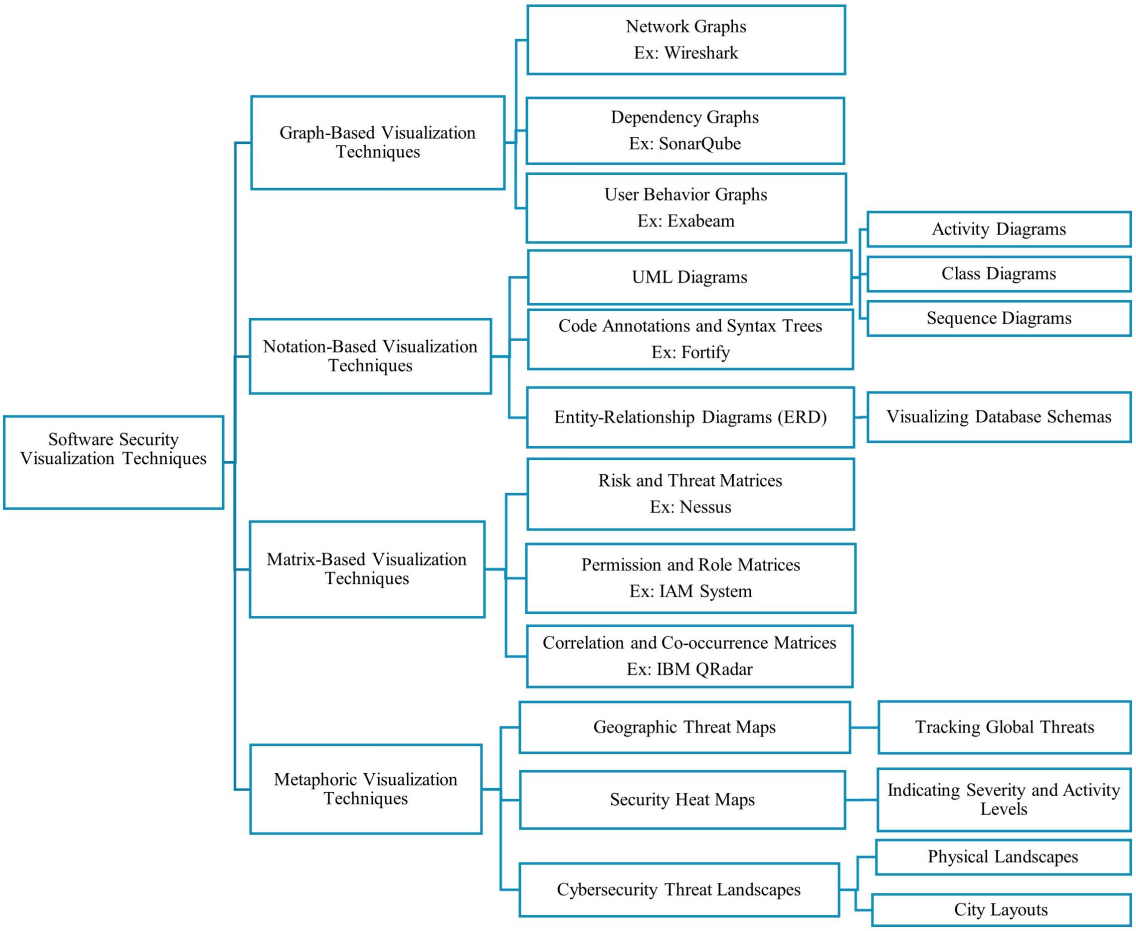


Fig. 2. Software security visualization techniques and tools.

application of dynamic analysis in visualizing security structures and properties of software systems, providing real-time security assessments [34]. While automation holds promise, its effectiveness in reducing human error and improving response times is still under debate, particularly in complex, rapidly evolving threat landscapes.

As shown in Fig. 2, this review explores various software security visualization techniques essential for understanding and managing security risks in complex systems. These techniques are categorized into four main types: Graph-Based, Notation-Based, Matrix-Based, and Metaphoric Visualization. Each category uses different techniques and tools to represent security data, helping security professionals, developers, and non-technical users gain insights into system vulnerabilities, threats, and user behaviors.

1) Graph-based visualization technique: Graph-based visualization techniques focus on visualizing connections, data flows, and dependencies within software systems to identify relationships, detect anomalies, and monitor user behavior [35]. Tools such as Wireshark are used for network graphs [36], SonarQube for dependency graphs [37], and Exabeam for user behavior graphs [38]. These techniques offer real-time insights into network traffic, application dependencies, and user interactions, crucial for detecting security threats in large systems.

2) Notation-based visualization techniques: Notation-based visualization techniques visually represent the design and architecture of software systems, helping developers understand code structure and the flow [39], [40]. Techniques like UML diagrams (including activity, class, and sequence diagrams) [41], tools like Fortify for code annotations and syntax trees [42], and Entity-Relationship Diagrams (ERD) are used to visualize database schemas [43]. This technique uses standardized diagrammatic representations to show the design and architecture of software systems and is useful in software design, debugging, and security risk assessment by mapping system components and their interactions.

3) Matrix-based visualization techniques: Matrix-based visualization techniques organize data into matrices to assess risks, permissions, and correlations within systems, offering a clear overview of user roles and access permissions [39]. Tools such as Nessus for risk and threat matrices [44], IAM Systems for permission, Role Matrices [45], and IBM QRadar for Correlation and Co-occurrence Matrices to assess security risks and map user roles [46]. These metrics are valuable in enterprise security, ensuring compliance, and detecting unauthorized access.

4) Metaphoric visualization techniques: Metaphoric visualization techniques use real-world metaphors, like geographic maps and city layouts, to intuitively represent complex security

data [47]. Geographic Threat Maps are used for tracking global threats [48], Security Heat Maps for indicating severity and activity levels [17], and Cybersecurity Threat Landscapes use physical and city layouts to represent security data [49], [39]. These techniques help non-technical users understand security risks and threats, improving situational awareness and aiding in action prioritization.

IV. Software Security Visualization in Practice

Software Security Visualization in Practice involves the application of visual techniques to enhance the understanding, monitoring, and management of security within software systems. As cybersecurity threats grow in complexity, traditional methods of analyzing security data often fall short. Visualization techniques bridge this gap by transforming complex data into intuitive, visual formats, enabling quicker and more accurate detection, analysis, and response to security threats. This practical approach to software security is critical for improving both the effectiveness and efficiency of cybersecurity efforts in real-world environments.

A. Operational Security Visualization

Operational security visualization is a crucial aspect in the field of cybersecurity, and it focuses on the graphical representation of security-related data to analyze security incidents. This study involves the use of visualization techniques and tools to interpret complex datasets, identify vulnerabilities, and maintain the integrity of the system. Fig. 3 illustrates Operational Security Visualization, which includes techniques for analyzing log files and SIEM data to monitor and enhance security operations.

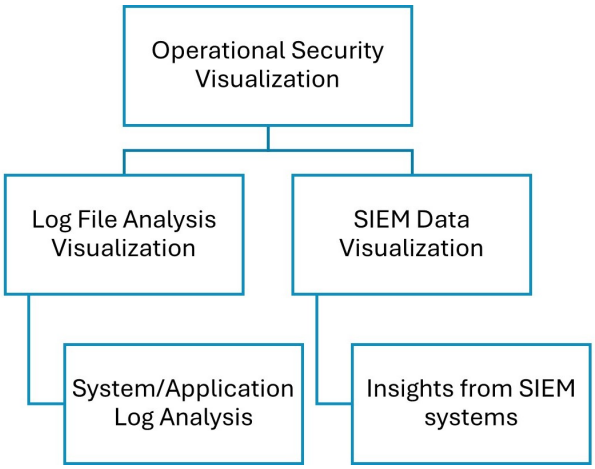


Fig. 3. Operational security visualization.

1) Log file analysis visualization: Log file analysis visualization helps in identifying security vulnerabilities, understanding system behavior, and enhancing overall security posture. Recent studies have emphasized the importance of visualization techniques to manage and interpret network logs, categorizing various security visual analytics methods, and highlighting the challenges of extracting useful information from raw log data.

One study discussed how the process of visualizing cyber-physical systems involves transforming raw unstructured data into information that can be effectively represented and analyzed, underscoring the necessity of robust visualization tools in cybersecurity [7]. However, this study did not extensively compare the effectiveness of different visualization tools, leaving a gap in understanding which tools are most effective for specific tasks. Another study by Korzeniowski and Goczyła highlighted recent advancements in automated log analysis, noting the benefits of standardizing log data formats to improve real-time monitoring, system health understanding, and root cause analysis [50]. Despite these advancements, integrating standardized log formats into existing security infrastructures remains a challenge, particularly in large-scale systems. The role of log file analysis in cybersecurity assessments is emphasized, reviewing methodologies and tools for measuring and assessing enterprise security, identifying vulnerabilities, and improving overall security infrastructure [51]. While these methodologies are effective, their scalability and applicability across different types of enterprises are still underexplored, representing a significant research gap.

2) SIEM data visualization: SIEM data visualization plays a crucial role in interpreting and managing the extensive amounts of security-related data collected by Security Information and Event Management (SIEM) systems. Recent research underscores the integration of big data analytics with SIEM to enhance the analysis and visualization of network security threats.

For instance, a study on effective security monitoring using efficient SIEM architecture emphasizes the use of advanced big data analytics tools to improve threat detection and visualization capabilities [52]. Additionally, another study highlights the limitations of traditional SIEM tools, such as their struggle to handle large-scale data efficiently and the inability to detect sophisticated attacks. This research highlights the need for a holistic framework that incorporates advanced visualization techniques to boost threat detection and response, addressing the challenges of continuous monitoring and rapid analysis of substantial volumes of security data in modern enterprises [53].

Another study focused on cyber situational awareness (CSA) visualizations, underscoring the need for sophisticated visualization techniques to support various stakeholders and improve threat understanding and response planning [9]. Despite these findings, the challenge of tailoring these visualizations to different user levels, particularly non-expert users, remains underexplored, presenting another gap in the current literature. Additionally, recent research has addressed the challenges of processing unstructured text logs in SIEM systems, emphasizing the importance of visualization in making sense of this data to detect patterns and anomalies [54].

These studies collectively demonstrate that effective visualization techniques are essential for improving the functionality and effectiveness of SIEM systems, enabling better threat detection, situational awareness, and overall cybersecurity. However, further research is needed to develop more advanced, user-friendly visualization tools that can handle the increasing complexity and volume of security data.

B. Cyber Security Visualization

Cybersecurity is a significant component of modern cybersecurity strategies, transforming complex data into visual

formats that facilitate cyber threat understanding, detection, and mitigation. Fig. 4 depicts Cybersecurity Visualization, which includes techniques for visualizing threat intelligence, phishing attacks, and malware behavior patterns to enhance security analysis and response.

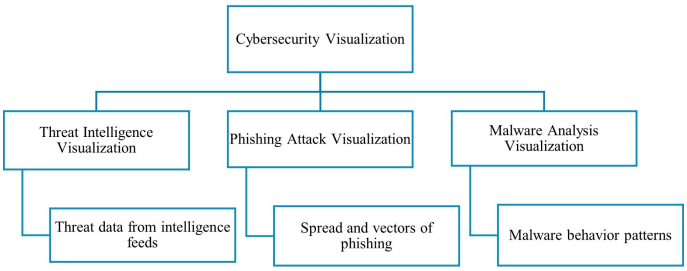


Fig. 4. Cyber security visualization.

1) Threat intelligence visualization: Threat intelligence visualization is vital for enhancing software security by transforming complex threat data into actionable insights. A recent study has emphasized the need for visualizing threat data from surface, deep, and dark web levels to aid in cybercrime detection and support law enforcement agencies [55]. Sun et al. highlighted the importance of real-time threat data visualization for proactive cybersecurity defense, providing a comprehensive review of recent research on cyber threat intelligence mining [18]. While these studies underscore the importance of visualization, they often overlook the practical challenges of implementing these techniques in real-world scenarios, such as the need for real-time processing and integrating multiple data sources.

A study has discussed the use of natural language processing techniques to extract cyber threat intelligence from unstructured texts, demonstrating how visualization can help interpret large volumes of data and identify key indicators of compromise. It also highlighted the automation of CTI extraction and the role of visualization in supporting proactive decision-making and threat mitigation [56]. However, the scalability of these techniques remains a concern, particularly in large organizations with vast amounts of unstructured data. Recent studies have demonstrated the need for standardized visualization techniques to improve the quality and sharing of technical threat intelligence, facilitating better collaboration and response to cyber threats [18]. Despite these advances, the lack of universally accepted standards for threat intelligence visualization continues to hinder effective collaboration between organizations. Collectively, these studies underscore the critical role of visualization in understanding and responding to cyber threats, thereby enhancing the overall security posture of organizations.

2) Phishing attack visualization: Phishing attack visualization is an essential tool in enhancing cybersecurity by transforming complex data into actionable insights to detect and mitigate phishing threats. Recent studies have underscored the significant role of software security visualization in the real-time detection and monitoring of phishing attacks. For instance, recent research highlights the importance of utilizing big data analytics within software security visualization to decrease the number of phishing attacks, emphasizing that visualizing large datasets can expose malicious patterns and enhance threat detection [57]. However, while these techniques are effective, their reliance on large data sets can also be a limitation, particularly in environments where data availability is restricted.

The effectiveness of machine learning-based visualization techniques in detecting phishing activities is highlighted, offering clear visual cues to identify suspicious behaviors [58]. Patil and Arra focused on the role of visual tools in user awareness training, demonstrating how visualizing phishing threats can educate users and improve their response to such attacks [59]. Despite these advancements, the challenge of integrating these tools into existing training programs and ensuring their accessibility to non-technical users remains underexplored. Another recent study provided a comprehensive review of cybersecurity measures, emphasizing the need for visualization in understanding and controlling phishing attacks [60]. These studies collectively underscore the critical role of visualization in enhancing the detection, prevention, and response to phishing threats, thereby improving the overall security posture of organizations. However, further research is needed to improve the usability and accessibility of these tools across different organizational contexts.

3) Malware analysis visualization: Malware analysis visualization is crucial in the fight against sophisticated cyber threats, offering a visual approach to understanding and mitigating malicious activities. A study highlighted the importance of Cyber Situational Awareness (CSA) in comprehending complex cyber threats, emphasizing the role of visualization in providing operational-level insights but noting gaps in visualizations for higher-level decision-making [17]. Another study has provided a systematic review of malware visualization techniques, showcasing their effectiveness in detecting, classifying, and identifying malware through static data visualization and network traffic monitoring [61]. Despite these findings, the study noted that the effectiveness of these techniques in large-scale, real-world environments still needs to be validated. Moawad et al. highlighted visualization-based malware detection, noting its efficiency and independence from domain experts, using image representation of malware binaries to enhance detection accuracy [62]. The challenge of maintaining accuracy and reducing false positives in these visualizations remains a significant barrier to their broader adoption.

Recent studies have emphasized the importance of advanced visualization systems in malware analysis. One such study conducted a comprehensive survey on the integration of explainable AI (XAI) methods for malware hunting. This research evaluated various XAI techniques for their effectiveness in visualizing and analyzing malware [63]. However, the challenge of integrating these advanced visualization techniques into existing malware analysis workflows remains a significant area for future research. These studies collectively underscore the significant role of visualization in malware analysis, enhancing the detection, classification, and mitigation of malware by transforming complex data into understandable visual formats, thereby supporting both operational and strategic cybersecurity efforts.

V. Discussion

This systematic review has provided an in-depth exploration of the current state of software security visualization

techniques. Key findings indicate that visualization techniques are crucial in improving security monitoring, threat detection, and overall system understanding. The review categorized the main approaches into four types: graph-based, notation-based, matrix-based, and metaphor-based visualizations.

Graph-based visualization techniques, such as network graphs (Wireshark) and user behavior graphs (Exabeam), are primarily effective in depicting connections, data flows, and system dependencies. These visualizations are particularly useful for real-time monitoring of dynamic systems, helping detect anomalies in user behavior and network traffic. However, they face challenges in scalability and real-time integration, particularly in large, complex systems. Notation-based visualization methods, like UML diagrams and entity-relationship diagrams (ERD), provide clarity on system design and architecture. They are instrumental in mapping out system components and flows, which is essential for security assessments. However, these tools may require significant effort from developers and may not adequately capture dynamic security events, limiting their effectiveness in real-time threat detection. Matrix-based visualization, used by tools like Nessus and IBM QRadar, focuses on assessing risks, permissions, and correlations within a system. These visualizations provide a structured view of user roles, permissions, and identified threats, making them effective for managing large-scale enterprise security environments. However, their static nature and reliance on predefined matrices can limit their adaptability to real-time threats, especially in rapidly evolving attack scenarios. Metaphor-based visualization techniques, such as security heat maps and geographic threat maps, leverage real-world metaphors to represent complex security data. These methods are especially useful in communicating security risks to non-technical stakeholders, improving overall situational awareness. However, their reliance on simplified representations may obscure the complexities of security systems, leading to oversights in more nuanced threat environments.

A. Recent Approaches

Recent research has introduced several innovative approaches to enhance software security visualization. One of the emerging trends is the integration of big data analytics and machine learning with visualization techniques. This allows for more dynamic threat detection, particularly in Security Information and Event Management (SIEM) systems. SIEM platforms, when combined with advanced visualization tools, can handle larger volumes of data more efficiently, providing real-time insights into complex security landscapes. Furthermore, the application of deep learning to automate vulnerability detection in static and dynamic analysis environments has gained attention. By applying machine learning algorithms to large datasets, these approaches can enhance the accuracy of vulnerability detection, though challenges remain in reducing false positives and improving the interpretability of results. In addition to machine learning-driven approaches, recent research has explored adaptive visualization mechanisms such as Level of Detail (LoD) techniques integrated with city metaphors to dynamically adjust visual complexity based on user roles, tasks, and real-time security data, thereby reducing cognitive overload and improving vulnerability detection [64].

B. Hybrid Visualization Techniques

An emerging concept in the literature is using hybrid visualization techniques, combining the strengths of multiple approaches to provide a more comprehensive view of security risks. For example, integrating graph-based visualizations with matrix-based approaches enables a holistic view of system structure and risk correlations. This hybrid approach provides deeper insights into security vulnerabilities, offering real-time mapping of user roles, permissions, and system behavior. However, the practical implementation of hybrid techniques is still under research, with scalability and performance in large-scale environments being key challenges.

C. Challenges

While current software security visualization tools offer significant benefits, several ongoing challenges persist. A major concern is the need for scalable solutions capable of processing large datasets in real-time, as modern software systems generate vast amounts of data that require real-time analysis [1]. Additionally, improving user accessibility is crucial. Many current tools are designed for experts, making them less intuitive for non-expert users, which hinders broader adoption [2], [19]. Enhancing user-centric design could make these tools more accessible and effective for a wider audience.

Another challenge is the lack of standardized metrics. The absence of consistent methods for assessing visualization performance makes it difficult to compare tools and measure their effectiveness across different studies [19]. Establishing standardized benchmarks could help address this issue and improve the reliability of security visualization tools.

VI. Conclusion

This review highlights the importance of visualization techniques in understanding, analyzing, and mitigating security threats by transforming complex data into easily interpretable formats. Dynamic analysis visualization supports detecting vulnerabilities in real time and also provides clear visuals that improve Cyber Situational Awareness (CSA) for everyone, especially non-experts, by focusing on threat information. However, several key challenges persist in the field. These include the need for scalable tools capable of processing large datasets in real-time as software systems grow in complexity, the lack of user-centric designs that hinder the accessibility of these tools to non-technical users, and the absence of standardized metrics to assess the effectiveness of visualization tools. These advancements will lead to more reliable and accessible software security solutions, ultimately enhancing the ability to defend against emerging security threats.

Future research should focus on integrating machine learning, Artificial Intelligence, and real-time data analysis capabilities while designing comprehensive platforms that unify various visualization techniques to enhance software security.

References

[1] L. Cui, J. Cui, Z. Hao, L. Li, Z. Ding, and Y. Liu, "An empirical study of vulnerability discovery methods over the past ten years," *Computers & Security*, vol. 120, p. 102817, 2022.

[2] L. Braz, C. Aeberhard, G. Çalikli, and A. Bacchelli, "Less is more: Supporting developers in vulnerability detection during code review," 2022.

[3] A. Abeyratne, C. Samarage, B. Dahanayake, C. Wijesiriwardana, and P. Wimalaratne, "A security specific knowledge modelling approach for secure software engineering," *Journal of the National Science Foundation of Sri Lanka*, vol. 48, no. 1, 2020.

[4] J. Senanayake, H. Kalutarage, M. O. Al-Kadri, A. Petrovski, and L. Piras, "Android source code vulnerability detection: a systematic literature review," *ACM Computing Surveys*, vol. 55, no. 9, pp. 1–37, 2023.

[5] C. Wijesiriwardana and P. Wimalaratne, "On the detection and analysis of software security vulnerabilities," in *2017 International Conference on IoT and Application (ICIOT)*. IEEE, 2017, pp. 1–4.

[6] J. Mortara, P. Collet, and A.-M. Dery-Pinna, "Visualization of object-oriented variability implementations as cities," in *2021 IEEE Working Conference on Software Visualization (VISSOFT)*, 2021.

[7] V. Cobilean *et al.*, "A review of visualization methods for cyber-physical security: Smart grid case study," *IEEE Access*, 2023.

[8] M. Siavvas *et al.*, "Security monitoring during software development: An industrial case study," *Applied Sciences*, vol. 13, no. 12, 2023.

[9] M. J. Page *et al.*, "The prisma 2020 statement: An updated guideline for reporting systematic reviews," *BMJ*, 2021.

[10] C. Okoli, "A guide to conducting a standalone systematic literature review," http://aisel.aisnet.org/cais, 2015.

[11] B. Kitchenham and P. Brereton, "A systematic review of systematic review process research in software engineering," *Information and Software Technology*, vol. 55, no. 12, pp. 2049–2075, 2013.

[12] H. Snyder, "Literature review as a research methodology: An overview and guidelines," *Journal of Business Research*, vol. 104, pp. 333–339, 2019.

[13] P. Brereton, B. A. Kitchenham, D. Budgen, M. Turner, and M. Khalil, "Lessons from applying the systematic literature review process within the software engineering domain," *Journal of Systems and Software*, vol. 80, no. 4, pp. 571–583, 2007.

[14] B. Kitchenham, "Guidelines for performing systematic literature reviews in software engineering," https://www.researchgate.net/publication/302924724, EBSE, Tech. Rep., 2007.

[15] Z. Munn, M. D. J. Peters, C. Stern, C. Tufanaru, A. McArthur, and E. Aromataris, "Systematic review or scoping review? guidance for authors when choosing between a systematic or scoping review approach," *BMC Medical Research Methodology*, vol. 18, no. 1, 2018.

[16] N. Chotisarn *et al.*, "A systematic literature review of modern software visualization," *Journal of Visualization*, 2020.

[17] L. Jiang, A. Jayatilaka, M. Nasim, M. Grobler, M. Zahedi, and M. A. Babar, "Systematic literature review on cyber situational awareness visualizations," *IEEE Access*, 2022.

[18] N. Sun *et al.*, "Cyber threat intelligence mining for proactive cybersecurity defense: A survey and new perspectives," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 3, pp. 1748–1774, 2023.

[19] V. Möllerström and J. Roos, "Engineering degree project code review application—simplifying code review through data flow visualization," 2023.

[20] A. Y. Wong, G. Chekole, M. Ochoa, and J. Zhou, "On the security of containers: Threat modeling, attack analysis, and mitigation strategies," https://www.exploit-db.com/, 2023.

[21] P. Zeng, G. Lin, L. Pan, Y. Tai, and J. Zhang, "Software vulnerability analysis and discovery using deep learning techniques: A survey," *IEEE Access*, 2020.

[22] R. L. Alaoui and N. E. H., "Deep learning for vulnerability and attack detection on web applications: A systematic literature review," *Future Internet*, vol. 14, no. 4, p. 118, 2022.

[23] E. Fedorchenko *et al.*, "An analytical review of the source code models for exploit analysis," *Information*, vol. 14, no. 9, 2023.

[24] V. Piantadosi, G. Rosa, D. Placella, S. Scalabrino, and R. Oliveto, "Detecting functional and security-related issues in smart contracts: A systematic literature review," 2023.

[25] S. Sakharkar, "Systematic review: Analysis of coding vulnerabilities across languages," *Journal of Information Security*, vol. 14, no. 4, pp. 330–342, 2023.

[26] E. B. Sørensen, "A literature review and practitioner survey on using vulnerability detection tools to defend against access control vulnerabilities," 2019.

[27] T. Cerny, A. S. Abdelfattah, V. Bushong, A. A. Maruf, and D. Taibi, "Microservice architecture reconstruction and visualization techniques: A review," *arXiv preprint arXiv:2207.02988*, Jul. 2022. [Online]. Available: http://arxiv.org/abs/2207.02988

[28] G. Parker *et al.*, "Visualizing anti-patterns in microservices at runtime: A systematic mapping study," *IEEE Access*, vol. 11, pp. 23 217–23 234, 2023.

[29] R. Raducu, R. J. Rodriguez, and P. Alvarez, "Defense and attack techniques against file-based toctou vulnerabilities: A systematic review," *IEEE Access*, vol. 10, pp. 22 758–22 775, 2022.

[30] R. Medeiros, J. Martinez, O. Díaz, and J. R. Falleri, "Visualizations for the evolution of variant-rich systems: A systematic mapping study," *Information and Software Technology*, vol. 154, Feb. 2023.

[31] E. H. Korkut and E. Surer, "Visualization in virtual reality: A systematic review," *Virtual Reality*, vol. 27, no. 2, pp. 1447–1480, Jan. 2023.

[32] K. Eberhard, "The effects of visualization on judgment and decision-making: a systematic literature review," *Management Review Quarterly*, vol. 73, no. 1, pp. 167–214, Feb. 2023.

[33] J. P. dos Reis, F. B. e Abreu, G. de F. Carneiro, and C. Anslow, "Code smells detection and visualization: A systematic literature review," *Archives of Computational Methods in Engineering*, vol. 29, no. 1, pp. 47–94, Dec. 2022.

[34] M. A. Lozano, I. P. Llopis, and M. E. Domingo, "Threat hunting architecture using a machine learning approach for critical infrastructures protection," *Big Data and Cognitive Computing*, vol. 7, no. 2, Jun. 2023.

[35] S. Lagraa, H. Mebrek, S. Houimli, V. Sivaraman, S. Radhakrishnan, and O. M. Lagraa, "A review on graph-based approaches for network security monitoring and botnet detection," *International Journal of Information Security*, vol. 23, no. 1, pp. 119–140, 2024.

[36] S. Arvind, A. Suresh, S. V. Kumar, P. Gopinath, N. Prakash, and P. R. Arvind, "Network traffic virtualization using wireshark and google maps," 2023.

[37] J. P. Biazotto, D. Feitosa, P. Avgeriou, and E. Y. Nakagawa, "Technical debt management automation: State of the art and future perspectives," *Information and Software Technology*, vol. 169, Mar. 2024.

[38] G. Sharma, A. Thakur, and C. Tiwari, "Developing a comprehensive framework for user and entity behavior analytics (ueba): Integrating advanced machine learning and contextual insights," *Journal of Communication Engineering & Systems*, vol. 14, no. 2, pp. 20–32, Jun. 2024.

[39] C. Wijesiriwardana, P. Wimalaratne, T. Abeysinghe, S. Shalika, N. Ahmed, and M. Mufarrij, "Software engineering secure codecity: 3-dimensional visualization of software security facets," *Journal of the National Science Foundation of Sri Lanka*, vol. 51, no. 3, pp. 437–450, 2023.

[40] N. Sinhabahu, P. Wimalaratne, and C. Wijesiriwardana, "Secure codecity with evolution: Visualizing security vulnerability evolution of software systems," in *2020 20th International Conference on Advances in ICT for Emerging Regions (ICTer)*. IEEE, 2020, pp. 1–2.

[41] N. P. Mule, "Implementation and evaluation of software architecture visualization," 2023, [Online]. Available: www.bth.se.

[42] Aravind *et al.*, "Enhancing code security with codesheriff: A multi-language vulnerability scanner," Research Square, 2024.

[43] S. Fluchs, R. Drath, and A. Fay, "Evaluation of visual notations as a basis for ics security design decisions," *IEEE Access*, vol. 11, pp. 9967–9994, 2023.

[44] X. Zhao, Y. Zhang, J. Xue, C. Shan, and Z. Liu, "Research on network risk evaluation method based on a differential manifold," *IEEE Access*, vol. 8, pp. 66 315–66 326, 2020.

[45] V. Engström, P. Johnson, R. Lagerström, E. Ringdahl, and M. Wällstedt, "Automated security assessments of amazon web services environments," *ACM Transactions on Privacy and Security*, vol. 26, no. 2, Mar. 2023.

[46] B. Nour, M. Pourzandi, and M. Debbabi, "A survey on threat hunting in enterprise networks," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 4, pp. 2299–2324, 2023.

[47] D. Moreno-Lumbreras, J. M. Gonzalez-Barahona, G. Robles, and V. Cosentino, "The influence of the city metaphor and its derivates in software visualization," *Journal of Systems and Software*, vol. 210, Apr. 2024.

[48] B. Jiang, X. You, K. Li, T. Li, X. Wang, and D. Si, "Virtual geo-cyber environments: metaphorical visualization of virtual cyberspace with geographical knowledge," *International Journal of Digital Earth*, 2024.

[49] D. G. Rosado *et al.*, "Managing cybersecurity risks of cyber-physical systems: The marisma-cps pattern," *Computers in Industry*, vol. 142, Nov. 2022.

[50] L. Korzeniowski and K. Goczyla, "Landscape of automated log analysis: A systematic literature review and mapping study," *IEEE Access*, vol. 10, pp. 22 435–22 457, 2022.

[51] S. F. Aboelfotoh and N. A. Hikal, "A review of cyber-security measuring and assessment methods for modern enterprises," *JOIV International Journal on Informatics Visualization*, vol. 3, no. 2, pp. 157–176, 2019.

[52] M. Sheeraz *et al.*, "Effective security monitoring using efficient siem architecture," *Human-centric Computing and Information Sciences*, vol. 13, 2023.

[53] A. Jose, "A holistic framework for unifying data security and management in modern enterprises," https://www.researchgate.net/publication/380153247, 2023.

[54] S. Reddy and P. Smiledirectclub, "Revolutionizing siem with ml-driven risk assessment and prioritization," *International Journal of Information Technology*, 2023.

[55] G. Cascavilla, D. A. Tamburri, and W. J. V. D. Heuvel, "Cybercrime threat intelligence: A systematic multi-vocal literature review," *Computers & Security*, vol. 107, p. 102258, 2021.

[56] M.-R. Rahman and W.-L. Rahman, "A literature review on mining cyberthreat intelligence from unstructured texts," *IEEE Access*, vol. 8, 2020.

[57] M. Pejic-Bach, I. Jajic, and T. Kamenjarska, "A bibliometric analysis of phishing in the big data era: High focus on algorithms and low focus on people," in *Procedia Computer Science*, vol. 218, 2023, pp. 91–98.

[58] S. Alnemari and M. Alshammari, "Detecting phishing domains using machine learning," *Applied Sciences (Switzerland)*, vol. 13, no. 8, 2023.

[59] K. Patil and S. R. Arra, "Detection of phishing and user awareness training in information security: A systematic literature review," in *Proc. 2nd Int. Conf. Innovative Practices in Technology and Management (ICIPTM)*, 2022.

[60] A. Verma and C. Shri, "Cyber security: A review of cyber crimes, security challenges and measures to control," *Vision*, vol. 29, no. 4, 2025.

[61] A. Alfagi, A. A. Manaf, A. Zainol, and A. A. Alarood, "Malware visualization technique: A systematic review," *Journal of Theoretical and Applied Information Technology*, vol. 15, no. 21, 2019, www.jatit.org.

[62] A. Moawad, A. I. Ebada, and A. M. Al-Zoghby, "A survey on visualization-based malware detection," *Journal of Cyber Security*, vol. 4, no. 3, pp. 169–184, 2022.

[63] M. Saqib, S. Mahdavifar, B. C. M. Fung, and P. Charland, "A comprehensive analysis of explainable ai for malware hunting," *ACM Computing Surveys*, 2024.

[64] I. Devendra, C. Wijesiriwardana, and P. Wimalaratne, "Enhancing software security visualization through adaptive level of detail (lod) mechanisms: A city metaphor-based approach," in *Proc. 9th Int. Conf. Inf. Technol. Res. (ICITR)*, Dec. 2024, pp. 1–6.

Human Versus AI: A Comparative Study of Zero-Shot LLMs and Transformer Models Against Human Annotations for Arabic Sentiment Analysis

Dimah Alahmadi
Information Systems Department, Faculty of Computing and Information Technology
King Abdulaziz University, Jeddah, Saudi Arabia

***Abstract*—Accurate sentiment analysis in Arabic natural language processing (NLP) remains a complex task due to the language's rich morphology, syntactic variability, and diverse dialects. Traditional annotation approaches require human experts, face significant challenges related to inter-annotator agreement and dialectal understanding. Recent advances in transformer-based models and large language models (LLMs) offer new techniques to generate annotations. This paper presents a comparative evaluation of three sentiment annotation strategies applied to Saudi dialect tweets: human expert labeling, fine-tuned transformer models (specifically CAMeLBERT-DA), and zero-shot inference using GPT-4o. The selected CAMeLBERT-DA which is already trained specifically for Arabic sentiment tasks and dialects, demonstrates robust performance with fast, scalable predictions. On the other hand, the selected GPT-4o shows competitive zero-shot accuracy without fine-tuning, making it a practical solution for real-time applications. We investigate how each approach performs on two datasets, both of more than 4,000 Saudi tweets covering a wide spectrum of dialects and sentiment expressions. Our methodology involves analyzing consistency across annotations using inter-rater agreement metrics such as Cohen's Kappa, Pearson correlation, and class-specific agreement rates. The results reveal that while human annotations capture cultural and context subtleties, they suffer from inconsistency, particularly in ambiguous or dialect-specific cases. This study contributes to the growing body of work on annotation methodologies by highlighting the strengths and limitations of both human and AI-based annotators in Arabic NLP. Our findings suggest that the zero-shot use of domain-specific transformers like CAMeLBERT-DA with general-purpose LLMs such as GPT-4o have a moderate correlation compared to actual human annotators. The paper concludes with recommendations for building reliable ground truth datasets and integrating AI-assisted labeling into Arabic NLP tasks.**

Keywords*—Large Language Model (LLMs); transformers; NLP; annotation; inter agreement; sentiment analysis, Saudi dialects

I. Introduction

The rapid advancement of large language models (LLMs) has transformed natural language processing (NLP) across multiple languages, including Arabic. Despite this progress, Arabic remains underrepresented and presents unique challenges due to its complex morphology, diglossia (the co-existence of Modern Standard Arabic and diverse dialects), and wide regional variation [1] and [2]. These linguistic characteristics make high-quality annotation for Arabic NLP tasks particularly demanding. The worldwide use and exchange of data with diverse platforms facilitate large-scale interactions between individuals. This resulted in a volume of text data where NLP plays a critical role in the future of smart applications. The advancement in NLP techniques helps to maintain and analyze such a volume of text data to extract meaningful decisions similar to humans' processing in areas like machine translation, text summarization, question answering, and sentiment analysis.

Manual annotation by expert linguists is often considered the gold standard for creating reliable datasets. However, this process is resource-intensive, time-consuming, and difficult to scale—especially when applied to informal and diverse content like social media. In recent years, researchers have turned to large language models (LLMs), such as OpenAI's GPT series for different tasks in NLP, such as sentiment analysis and machine translation [3], GPTs have a potential solution to reduce the burden of manual annotation. These models, particularly in zero-shot and few-shot settings, have shown remarkable success in performing NLP tasks without domain-specific training, raising questions about their reliability as annotation tools.

In the context of Arabic NLP, the utility of LLMs for sentiment analysis and dialect-sensitive tasks has not been comprehensively evaluated. Additionally, there is a need to benchmark the quality of their outputs against those produced by human annotators. This is particularly crucial when LLMs are used to process social media data, where language use tends to be informal, ambiguous, and culturally nuanced [4].

This study investigates the effectiveness of zero-shot GPT-4o in predicting Arabic sentiment analysis and fine-tuned transformer models (e.g., CAMeLBERT-DA) ability compared to human expert annotations. We aim to explore the validity and practical trade-offs of each annotation methodology.

The research is guided by the following questions:

- How do GPT-4o and CAMeLBERT-DA perform in zero-shot sentiment classification of Saudi dialect tweets compared to human annotation?
- How do annotation metrics differ in terms of agreement and correlation for real-world Arabic dialect social media datasets?

By addressing these questions, the study contributes to the growing body of research on efficient informed annotation practices in low-resource and linguistically diverse settings such as Arabic NLP.

A. Literature Review

Recent studies have highlighted several research gaps in this domain. Notably, there is a lack of thorough investigation of the annotation quality produced by LLMs for Arabic NLP tasks, particularly across different dialects and task types. While some work has benchmarked LLM performance against traditional models [5]. In addition, few have directly compared LLM-generated labels to human-annotated gold standards in Arabic and examined sources of error in LLM-generated annotations and transformers, as well as where they diverge most from human judgments.

B. Annotation Challenges in Arabic NLP

Arabic presents unique annotation challenges in NLP due to its morphological complexity, rich syntax, and multiple dialects that deviate significantly from Modern Standard Arabic (MSA) [6]. Arabic is a polyglossic language comprising Modern Standard Arabic (MSA) and numerous regional dialects. These dialects differ significantly in phonology, morphology, lexicon, and syntax, making unified dataset construction particularly difficult [7]. Annotators must often be familiar with both MSA and dialectal varieties to accurately label sentiment, intent, or named entities, especially when working with informal texts like tweets or online forums.

Another key challenge lies in the scarcity of large-scale, publicly available annotated datasets. Unlike English, where benchmarks like GLUE and SuperGLUE have standardized evaluation protocols, Arabic NLP still suffers from fragmented resources [8]. Many datasets are restricted to specific genres such as news propaganda, limiting their utility in social or conversational contexts on X [9].

Moreover, the presence of ambiguous structures, mixed-code usage (e.g., Arabic-English), and sarcasm prevalent in social media further complicate the annotation process. These linguistic phenomena require not only syntactic understanding but also pragmatic and cultural awareness. Thus, high inter-annotator agreement is difficult to achieve, particularly when annotations are conducted by non-experts [10].

Another major challenge is the high-quality annotated datasets in Arabic. Many Arabic NLP tasks – from hate speech detection to sentiment analysis suffer from trust labeled data, especially compared to English [11]. This is partly due to the labor-intensive nature of creating Arabic corpora: manual annotation is time-consuming, expensive, and often requires domain or dialect experts. The process can be prone to inconsistencies as well; human annotators sometimes disagree on labels for complex or ambiguous texts, introducing variability into the dataset. Arabic's writing system (e.g. omission of short vowels) and extensive morphological ambiguity can further complicate annotation, since the correct interpretation of a word or sentence may rely on context or annotator intuition. In tasks like offensive language detection, the lack of diacritics and the prevalence of informal expressions make it hard for annotators to determine intent or meaning without context [12]. Authors in study [13] have presented blockchain technology to generate a wide trust annotated dataset by crowdsourcing applications that are based on blockchain structures.

In summary, the literature indicates that Arabic NLP faces persistent annotation challenges: scarcity of labeled data, dialect and domain mismatches between annotators and content, and the inherent linguistic complexity of Arabic. These issues need careful dataset design (with clear guidelines and training for annotators) and innovative approaches to support or complement human annotators, as we explore in subsequent sections.

C. Transformer and GPT Models for Arabic NLP

In response to these limitations, large language models (LLMs) have emerged as a promising alternative to manual annotation. Transformer-based models such as BERT [14] and T5 [15] have been adapted for Arabic via pretraining on Arabic corpora.

Transformer-based models have revolutionized Arabic NLP. Initial efforts like AraBERT and Arabic-BERT demonstrated strong performance gains by pretraining on large Arabic corpora [16], [17]. Later, MARBERT and ARBERT incorporated dialectal Arabic into their training to improve generalization across informal text [6]. These models consistently outperformed multilingual baselines on downstream tasks.

Notably, CAMeLBERT-DA [18] was trained on the CAMeL Lab's extensive Arabic text collections and fine-tuned for dialectal sentiment classification, showing state-of-the-art performance on multiple downstream tasks.

Meanwhile, multilingual and instruction-tuned models like mT5 [19] and XGLM [20] have shown competence in zero-shot tasks. These models, though not Arabic-specific, benefit from broad cross-lingual transfer. However, they still underperform on dialectal Arabic unless further adapted or fine-tuned.

With the release of ChatGPT and the GPT-4 series [21], zero-shot and few-shot prompting has gained traction. These models exhibit impressive ability to generalize across tasks, offering a practical solution for rapid annotation. However, their performance in Arabic sentiment annotation is still underexplored, especially in informal settings. Recent work by Khalifa et al. [22] highlighted that GPT-4 can approximate expert annotation on formal Arabic , but may struggle with dialectal variation and ambiguous expressions.

Transformer models have not been limited to encoders. Researchers also explored generative transformers for Arabic. For instance, an AraGPT2 model (an Arabic version of GPT-2) was developed as a language generator [23]. AraGPT2 has been used alongside BERT models in hybrid systems; one study combined CAMeLBERT (encoder) with AraGPT2 (generator) in an ensemble, achieving superior accuracy on an Arabic text classification task. However, compared to the proliferation of Arabic BERT models, generative pre-trained transformers dedicated to Arabic are fewer. This is changing with the rise of massive multilingual models like GPT-3 and GPT-4, which, despite not being Arabic-specific, are trained on enough Arabic text to demonstrate impressive capabilities in the language. These GPT-series models can perform Arabic tasks in a zero-shot or few-shot manner – a fundamentally different paradigm from task-specific fine-tuning. Researchers have been keen to evaluate how well such models handle Arabic understanding and generation.

In a comprehensive evaluation across 60+ Arabic NLP benchmarks, Khondaker et al. found that ChatGPT's zero-

shot accuracy was consistently lower than that of dedicated Arabic models on tasks like text classification, NER, and QA [24]. These models enable broader experimentation in Arabic NLP, but challenges remain in understanding their bias and limitations across dialects and domains. These findings suggest that while large GPT-style models are very powerful, Arabic-specific adaptation (through fine-tuning or prompting techniques) remains important for top performance. Nevertheless, the convenience of prompting an LLM without needing task-specific training is highly attractive, particularly for addressing the data scarcity issue. This has led to explorations of using GPT models as annotation tools, which we discuss in this research.

D. Hybrid Annotation Approaches

Hybrid annotation workflows are a growing area of interest. In this approach, LLMs generate initial labels that are then validated or corrected by human experts. Studies in English NLP [25] show this reduces cost and improves consistency. This framework remains under-investigated in Arabic NLP, despite its potential for scalable, culturally sensitive annotation. There is a pressing need for empirical studies that evaluate such workflows across diverse dialects, task types, and prompt strategies. One approach is to use LLMs to generate initial labels or data, which are then verified by humans. For example, Senator et al. employed ChatGPT to perform semantic role labeling (SRL) on Arabic texts. In their study, ChatGPT was prompted to assign semantic role tags to Arabic sentences (including emotional text), effectively acting as an annotator. Remarkably, in a cross-lingual setting – projecting English SRL annotations onto Arabic via ChatGPT's translation – then LLM achieved about 94% accuracy, approaching human quality [26].

In conclusion, the period 2020–2025 has seen rapid progress in both Arabic-specific NLP models and the use of large generative models, and the capabilities lie in combining these advances to develop robust, efficient annotation methodologies that leverage the best of human and machine intelligence. The literature review suggests that while LLMs offer valuable annotation support, their optimal use in Arabic NLP requires deeper investigation into prompt engineering, dialect adaptation, and comparison with human oversight. This study builds on these insights by systematically comparing human annotations, CAMeLBERT-DA predictions, and GPT-4o zero-shot outputs on Arabic tweets.

II. Methodology and Experimental Setup

This study explores whether AI models with zero-shot strategies can be used to build the ground truth. It also compares the results using the Cohen Kappa metric and the Pearson correlation coefficient. The analysis of humans vs. AI models in NLP tasks, specifically sentiment analysis, using two Saudi dialect datasets. Fig. 1 illustrates the steps in the general methodology.

A. Datasets

The following is a description of the datasets included in this study's experiments:

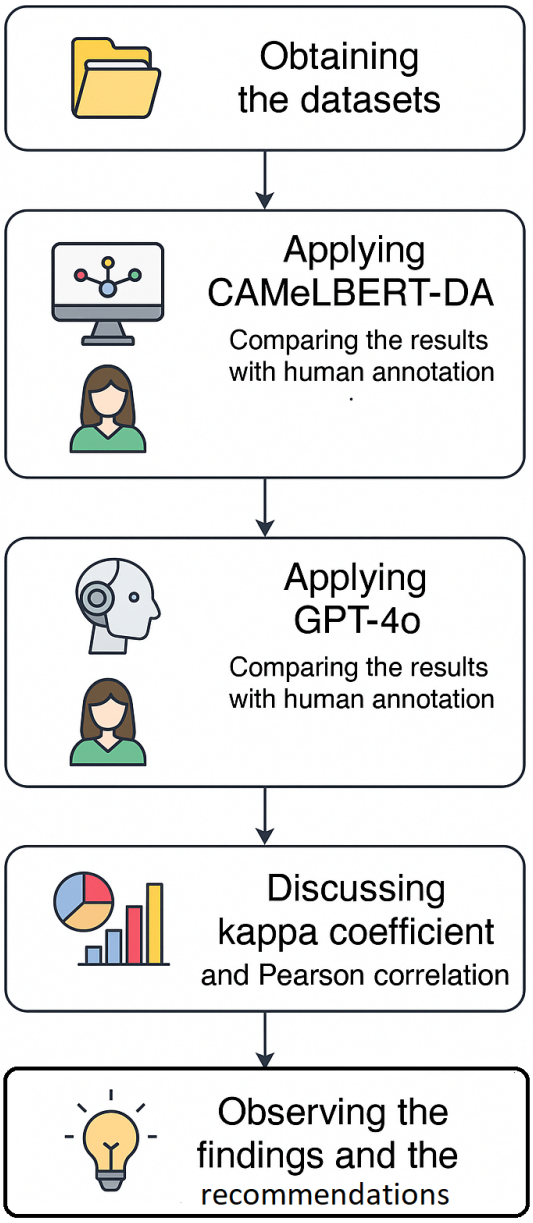


Fig. 1. Methodology overview.

1) SDCT: Multi-Dialects Corpus Classification for Saudi Tweets in [27]. It is dedicated to Saudi dialects. The data are from the X platform, as it is a rich source of data for short posts to be collected. The authors performed the pre-processing phases that involved main tasks, which are data cleaning and normalization. After data collection and pre-processing, the data were manually annotated by human experts with a total of 4181 records. In this study, we used the version that had removed advertisement tweets in the data cleaning phase. Also, it is prepared by removing links (http://), mentions (@Username), retweets, hashtags, and punctuation. While emojis were kept to enrich the sentiment detection.

2) Tb-SAC: Topic-based and Sentiment Classification for Saudi Dialects Tweets. It is extracted from corpora from X, especially from Saudi dialects. The corpus contains 4301 tweets, which are labeled based on sentiments using a common scale:

positive, negative, and neutral. First, preprocessing step by removing all the noise data in the tweets that were not related, such as mentions, hashtags, URLs, white space, retweets, punctuation marks, and redundant tweets. In the second step, normalization is the process of standardizing the form of some Arabic letters that have various shapes to be unified in one, but maintain the same meaning. The dataset was normalized by applying the Tashaphyne library and lemmatization. Emojis here are also removed; only the root words text was kept [28].

B. Models Selection

Based on the NLP task included in the experiment, which is sentiment analysis, the models used are selected. The selected transformer models have two criteria: fine-tuned and have achieved high accuracy in the domain of short tweets. And they consider recent research and benchmarks in the Arabic NLP tasks. The Table I below summarizes the top Arabic transformer models for sentiment analysis, dialect identification, which can be used (either as zero-shot classifiers or pre-trained fine-tuned models). All models are available on Hugging Face (HF). We highlight whether each supports zero-shot use or is a fine-tuned classifier, provide benchmark accuracy/F1 notes, and explain why it's recommended. CAMeL Lab's CAMeLBERT-DA Sentiment Analysis model, fine-tuned for Arabic sentiment classification (positive/negative). This model handles Modern Standard Arabic and dialectal content, as it was trained on diverse Twitter datasets (ASTD, ArSAS, SemEval). (Hugging Face: https://huggingface.co/CAMeL-Lab/bert-base-arabic-camelbert-da-sentiment) CAMeLBERT is a collection of BERT models pre-trained on Arabic texts with different sizes and variants. the release of pre-trained language models for Modern Standard Arabic (MSA), dialectal Arabic (DA), and classical Arabic (CA), in addition to a model pre-trained on a mix of the three [18]. Table I presents the reasons this study adopts this transformer, as many works support [29] and [30].

ChatGPT is an application utilizing the large artificial intelligence models GPT-3 or GPT-4 to generate text based on user input. GPT denotes a neural network adept at learning from extensive datasets to generate natural language outputs. GPT-4 is characterized as more creative, reliable, and collaborative than its predecessor GPT-3 [31] and [32]. The cutting-edge advancement in LLMs [33], GPT-4o was chosen for the comparison in the study experiment via an OpenAI API account for paying customers. It delivers state-of-the-art sentiment classification accuracy without dedicated task training, demonstrating competitive performance against domain-tuned transformer models. Zero-Shot Versatility and Ease of Use: GPT-4 is a large pre-trained model that can perform sentiment analysis without the need for task-specific fine-tuning or large labeled datasets. Broad Language Understanding & Nuance: As a massive foundation model, GPT-4 was trained on an extremely large and diverse corpus (including substantial Arabic content). Table II shows a comparison between the Arabic transformer CAMeLBERT-DA and the large language model GPT-4.

Regarding model structure, GPT-4 and CAMeLBERT-DA have different transformer architectures. GPT-4 (Generative Pre-trained Transformer 4) is built as a decoder-only Transformer, meaning it generates text by predicting one token after another (typical for GPT-style models). CAMeLBERT-DA, on the other hand, is based on BERT, which uses a Transformer encoder – it reads the entire input sequence bidirectionally to produce a contextualized representation. In practice, this means GPT-4 is designed for text generation (and can handle dialogue or any generative tasks), while CAMeLBERT-DA's architecture is designed for understanding text and classification (it outputs encoded features that are fed to a classifier layer) [34] and [35].

C. Evaluation

During the annotation process to build the ground truth for both datasets, human volunteers were involved. In order to validate the human opinions of the sentiment and the AI-generated sentiment, two metrics:

- Inter-model agreement metrics are also used to assess the reliability between annotators. The degree of agreement indicates how accurate the emotion labels assigned to the text. The calculation of the Kappa coefficient and its interpretation were based on the well-known reliability equations and measurements presented in [25]. The kappa statistic revealed the actual reliability of the categorical labels, accounting for the possibility of random agreement.
- Recent NLP studies have employed Pearson correlation to quantify annotator similarity on sentiment and emotion tasks. [36]. Pearson's r is a convenient metric to assess how consistently they apply the scale. (Notably, even annotator confidence scores – if each annotator gives a confidence level for each label.

In summary, Pearson correlation is very useful when treating labels as interval data – it captures how well annotators' scoring trends align – but it should be paired with traditional inter-rater agreement metrics to fully evaluate agreement on the actual labels and to ensure that high correlation indeed reflects reliable, reproducible annotation.

D. Results and Discussion

The experiment here went through the zero-shot settings for both models compared to the human annotators.

1) Generating sentiment using CAMeLBERT-DA transformer: First, using zero-shot and direct inference of the CAMeLBERT transformer to predict the sentiment for dataset1 and dataset2, in Hugging Face [37].

We need to explore the labels produced using the transformer-based model and those in the ground truth by human experts. A detailed comparison from Fig. 2 , the actual sentiment distribution is: negative: 901 (21.5%), neutral: 2506 (59.9%) and positive: 774 (18.5%). on the other hand, Predicted Sentiment Distribution by the transformer was negative: 2607 (62.4%), neutral: 601 (14.4%), and positive: 973 (23.3%). The inter-agreement using Cohen's Kappa Score: 0.2130, while the Pearson Correlation: 0.437. This indicates "fair agreement" between the actual and predicted sentiments. The score suggests that the prediction model performs better than

TABLE I. COMPARISON OF CAMELBERT-DA WITH OTHER ARABIC TRANSFORMERS FOR ARABIC SENTIMENT ANALYSIS

Aspect	CAMeLBERT-DA	Arabic T5 / MARBERT / Others	Recommendation
Accuracy on Arabic Tweets	˜92–93% accuracy on ArSAS, ASTD, and SemEval; fine-tuned on multiple Arabic tweet datasets	MARBERT: competitive but slightly lower; Arabic T5 not specialized or benchmarked for sentiment on tweets	CAMeLBERT-DA outperforms or matches best models due to targeted fine-tuning on diverse tweet datasets
Real-Time Inference Speed	BERT-base encoder model (110M params); efficient for batch and streaming inference	Arabic T5: large encoder–decoder; slower generation; MARBERT: similar size but more general-purpose	CAMeLBERT-DA offers faster, lower-latency inference than T5-like models; ideal for live tweet analysis
Suitability for Tweets	Pretrained on MSA + Dialectal Arabic + social media text; robust to slang, emojis, informal spelling	Arabic T5 trained more on formal MSA and QA-style tasks; MARBERT strong on dialects but not sentiment-specific	CAMeLBERT-DA is dialect-aware, slang-aware, and sentiment-optimized—making it perfect for social content

TABLE II. COMPARISON BETWEEN GPT-4 (CHATGPT) AND CAMELBERT-DA FOR ARABIC SENTIMENT ANALYSIS

Aspect	GPT-4 (ChatGPT)	CAMeLBERT-DA (Arabic BERT)
Model Type	Large Language Model (LLM); generative, decoder-only transformer; predicts next tokens (autoregressive).	Pretrained BERT-base encoder; bidirectional transformer trained for understanding Arabic. Fine-tuned for classification.
Scale (Parameters)	Very large (estimated 1T+ tokens and hundreds of billions of parameters; exact size not disclosed). Trained on multilingual data including Arabic.	Base-sized model (110M parameters). Pretrained on 5.8B Arabic words from dialectal, MSA, and classical Arabic sources.
Pretraining Data	Trained on diverse multilingual web data (books, articles, web pages). Strong in MSA and general Arabic.	Trained on large-scale Arabic corpora from social media and diverse dialects (CAMeL Lab).
Fine-Tuning for Sentiment Analysis	Not required. Performs sentiment classification in zero- or few-shot mode via prompting. No task-specific finetuning.	Required and completed. Fine-tuned on ASTD, ArSAS, and SemEval datasets. Dedicated sentiment prediction head.
Output Type	Text generation. Can provide label and reasoning if prompted. Very flexible output.	Class label output (e.g., “positive” / “LABEL_1”) with probability score. Not designed for explanatory responses.
Sentiment Accuracy on Arabic Tweets	F1-score ≈ 0.75–0.80 (zero- or few-shot). Competitive with fine-tuned models. Excels in MSA and structured dialectal text.	F1-score ≈ 0.72–0.76 (fine-tuned). High accuracy on dialectal Arabic tweets due to domain-specific tuning.
Zero/Few-Shot Support	Native. Easily supports both modes without additional data. Highly flexible.	Not applicable. Requires fine-tuning and training on task-specific data.
Best Use Case	Real-time, on-demand analysis in flexible environments (e.g., chat, APIs). Great for label and explanation.	High-throughput classification on Arabic social media data. Ideal for static datasets with known structure.

random chance, but there’s significant room for improvement. Correlation: 0.4377 - There’s a moderate positive correlation between the two columns, indicating some relationship but not a strong one.

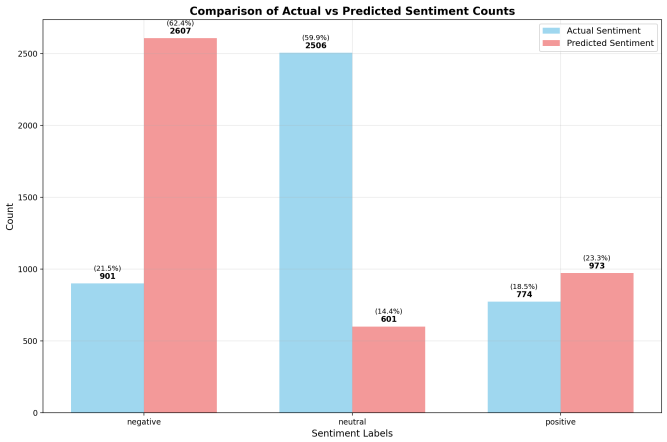


Fig. 2. Comparison of Humans vs. Predicted sentiment counts: (Dataset1).

Key observations:

- The actual sentiment data by humans is predominantly neutral (59.9%), while the predicted model tends to classify more samples as negative (62.4%)
- The model significantly under-predicts neutral sentiment (14.4% vs. 59.9% actual)
- The model over-predicts negative sentiment (62.4% vs. 21.5% actual)
- Positive sentiment predictions are closer to actual values (23.3% vs. 18.5%)

The confusion matrix in Fig. 3 shows where the model struggles most, particularly in correctly identifying neutral sentiments, often misclassifying them as negative. This suggests that positive sentiment is the most clear for humans and AI to detect and identify.

Now, the experiment of implementing the transformer for the second dataset. Key observations are found and perfect Alignment in some cases. The visualization in Fig. 4 shows that when the model is confident about its predictions, there’s perfect alignment between feeling and confidence labels. However, the presence of “unknown” labels for negative sentiments indicates areas where the model struggles with certainty, particularly for negative sentiment classification.

Key Insights from the Visualization:

- Perfect Prediction Accuracy for positive and neutral sentiments:
- All 2,086 actual positive sentiments were correctly predicted as positive.

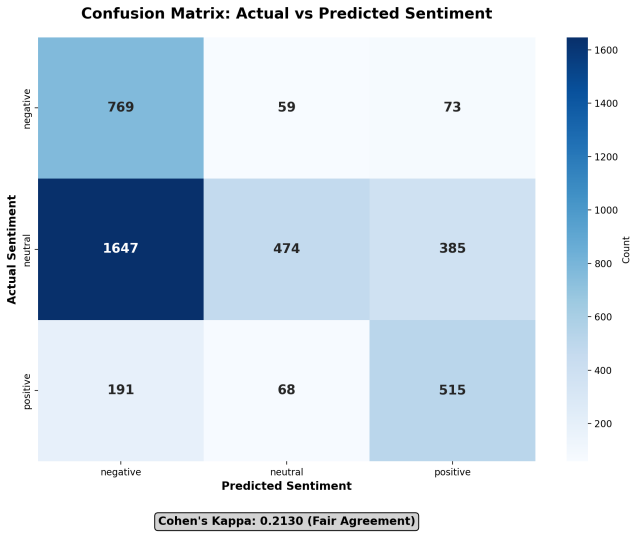


Fig. 3. Confusion matrix of Humans vs. Predicted sentiment labels with cohen's kappa score: (Dataset1).

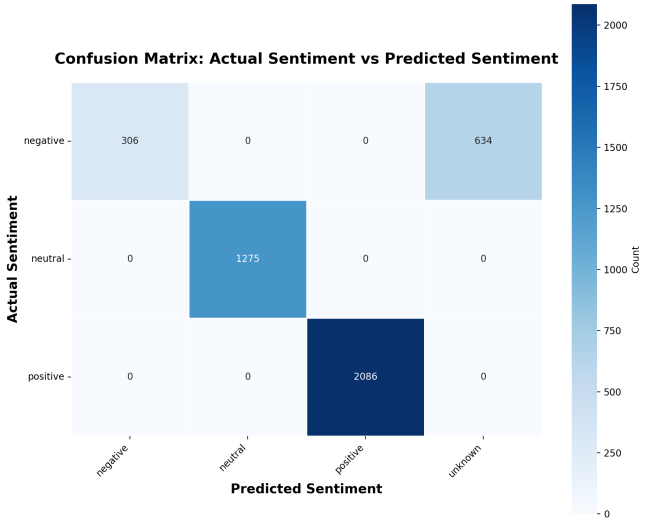


Fig. 5. Confusion matrix of actual Sentiment vs. Predicted sentiment including unknown class: (Dataset2).

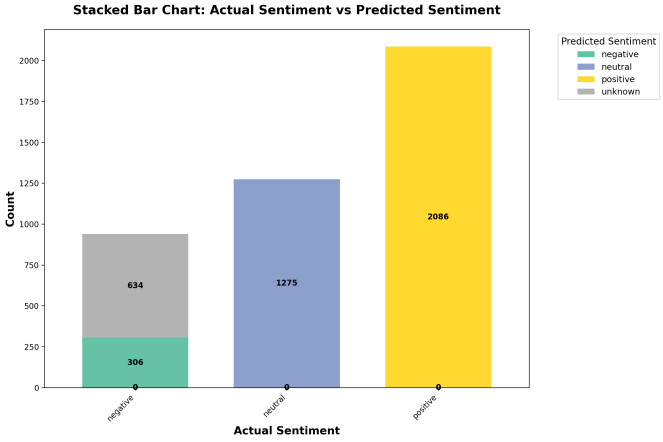


Fig. 4. Stacked bar chart of humans Sentiment vs. Predicted sentiment: (Dataset2).

- All 1,275 actual neutral sentiments were correctly predicted as neutral.
- Mixed Results for Negative Sentiment: Only 306 out of 940 actual negative sentiments were correctly predicted as negative 634 actual negative sentiments were predicted as "unknown", indicating model uncertainty.

The transformer here shows high confidence and accuracy for positive and neutral sentiments but struggles with negative sentiment classification, often defaulting to "unknown" when uncertain. The confusion matrix in Fig. 5 shows that the correlation coefficient between actual humans' sentiment and transformer prediction is 0.114, indicating a weak positive correlation. This suggests that while there is some relationship between the predicted feelings and confidence labels, it's not very strong.

2) GPT-4o: Using the chosen LLM from OpenAI represented in GPT-4o in this part of the experiment to validate the capacity of humans against AI. First of all, a clear, direct prompt based on Zero-Shot Prompt (No examples provided) is prepared as follows:

Analyze the following text for sentiment, and respond using only one of the three words: Positive, Negative, or Neutral.

Text: 'The service was excellent and very fast.'

Answer:

Positive

We analyzed the agreement of GPT-4o performance against human judgment for both dataset1 and dataset2. Table III shows the sentiment predicted labels using GPT-4o compared to human annotators in dataset1. The visualization shows GPT-4o has fair overall agreement (0.366) with humans but performs inconsistently across sentiment classes - excellent on neutral (81.4%), moderate on positive (63.2%), but poor on negative (39.3%) sentiment.

TABLE III. SENTIMENT ANNOTATION EVALUATION METRICS: GPT-4O VS. HUMANS (DATASET1)

Metric	Value	Interpretation
Cohen's Kappa	0.366	Fair agreement
Overall Agreement	58.0%	Moderate agreement
Pearson Correlation	0.557	Moderate correlation
Negative Agreement	39.3%	Poor agreement
Neutral Agreement	81.4%	Good agreement

On the other hand, an experiment using the second dataset was conducted. Table IV Inter-rater Agreement: The Cohen's Kappa score of 0.239 indicates a fair agreement between the human-labeled and the 'GPT-4o' sentiment classifications. This suggests that while there is some consistency, a notable degree of disagreement also exists. Overall Matching Discrepancy: Only 47.22% of the instances showed an exact match in sentiment labels between the two sources, highlighting a significant divergence in their overall classifications. Correlation

TABLE IV. SENTIMENT ANNOTATION EVALUATION METRICS: GPT-4O VS HUMANS (DATASET2)

Metric	Value	Interpretation
Cohen's Kappa	0.239	Fair agreement
Overall Agreement	47.22	Poor agreement
Pearson Correlation	0.433	Moderate correlation
Negative Agreement	54.46	Moderate agreement
Neutral Agreement	43.29	Poor agreement
Positive Agreement	45.60	Poor agreement

Strength: A moderate positive correlation (0.45) was observed between the numerical representations of the actual humans' feelings and 'GPT-4o' sentiments. This implies a general tendency for them to align, but not a strong linear relationship. Sentiment-Specific Consistency: Agreement levels vary across sentiment categories. Negative sentiment classifications show the highest specific agreement at 54.46%, suggesting better consistency for negative cases. Conversely, neutral sentiment exhibits the lowest specific agreement at 43.29%, indicating more challenges in aligning neutral classifications.

Finally, observations during the comparisons are as follows:

- The results show closer opinions of LLMs when the zero-shot setting is implemented and are more correlated to human experts' judgments. Transformers tend to show a moderate correlation with human annotators, while GPT-4o shows fair agreement. This is due to the fine-tuning setting of transformers for the specified domain to adapt the dataset.
- The efficiency and the cost of time using OpenAI for developers took 120 minutes to generate sentiment for every dataset, which counts around 4000 posts based on Google Colab using a paid account. On the other hand, the transformer CAMeLBERT-DA offline pipeline took only 60 seconds to generate the sentiment. This is because of the broad general use of LLMs and complexity of models and the number of parameters compared to the transformer, which is usually domain-specific.
- Another observation is the dataset level of preprocessing affects the AI decisions. In other words, during the experiment, the dataset2 version that had an extra level of preprocessing via lemmatization and removing all the emojis reduced the capability of models to detect the right sentiment, as it provides the lowest inter-agreement between GPT-4o (0.24) and CAMeLBERT-DA (0.14).

III. CONCLUSION AND FUTURE WORK

In this research, we compared OpenAI's GPT-4o model and transformer-based model against human experts for the annotation process. The experiments were conducted for NLP tasks in specific sentiment analysis prediction for Saudi dialect tweets. The finding revealed a better performance of GPT-4o against the transformer CAMeLBERT-DA. Inter-agreement Cohen Kappa was 0.366, indicating fair agreement between GPT-4o and human annotation for the first dataset, where initial preprocessing is conducted without any lemmatization, and emojis also remain in the text. Notably, datasets with intensive preprocessing produced difficulties in sentiment detection for transformers. This suggests that transformers and LLMs require more context to generate sentiment close to humans. The comparison between the models' performance and the human annotation explains that zero-shot manner in this experiment is fair, but not a completely reliable tool to substitute the human annotation process. For these reasons, sentiment generation research should investigate the combination of human crowdsourcing and automatic sentiment generation by LLMs. Furthermore, a limitation of this study is the lack of comparison across different Arabic dialects, such as Egyptian, Gulf, and Maghrebi. Also, in the future, employing few-shot strategies for LLMs and fine-tuned transformers can be studied for sentiment tasks and other NLP tasks such as Arabic text summarization. New evaluation metrics and empirical studies are critical to increasing reliability and trustworthiness when zero-shot methods are applied in real-time applications. In addition, the implications of fairness and bias issues can also be studied, which are crucial for ethical real-world deployment.

REFERENCES

[1] A. Elmadany, E. M. B. Nagoudi, and M. Abdul-Mageed, "Orca: A challenging benchmark for arabic language understanding," *arXiv preprint arXiv:2212.10758*, 2022.

[2] S. Aggarwal, B. Khanna, R. Madhumala, R. Garg, A. Maroju, and B. Brahma, "Analysis of arabic texts using semantics," in *International Conference on Computing and Communication Networks*. Springer, 2024, pp. 389–401.

[3] J. Algaraady and M. Mahyoob, "Exploring chatgpt's potential for augmenting post-editing in machine translation across multiple domains: challenges and opportunities," *Frontiers in Artificial Intelligence*, vol. 8, p. 1526293, 2025.

[4] G. Bourahouat, M. Abourezq, and N. Daoudi, "Systematic review of the arabic natural language processing: challenges, techniques and new trends," *Journal of Theoretical and Applied Information Technology*, vol. 101, no. 3, pp. 1333–1343, 2023.

[5] Z. Alyafeai, M. S. Alshaibani, B. AlKhamissi, H. Luqman, E. Alareqi, and A. Fadel, "Taqyim: Evaluating arabic nlp tasks using chatgpt models," *arXiv preprint arXiv:2306.16322*, 2023.

[6] M. Abdul-Mageed, A. Elmadany, and E. Nagoudi, "Arbert & marbert: Deep bidirectional transformers for arabic," *arXiv preprint arXiv:2101.01785*, 2020.

[7] A. Bourahouat *et al.*, "A survey on resources and tools for arabic nlp," *ACM Computing Surveys*, 2023.

[8] N. Habash *et al.*, "The state of arabic nlp: Resources and challenges," *Journal of Computational Linguistics*, 2022.

[9] B. M. Almotairy, M. Abdullah, and D. H. Alahmadi, "Dataset for detecting and characterizing arab computation propaganda on x," *Data in Brief*, vol. 53, p. 110089, 2024.

[10] A. Alharbi *et al.*, "Challenges in annotating dialectal arabic: An empirical study," in *Proceedings of LREC*, 2021.

[11] W. Zaghouani and M. R. Biswas, "An annotated corpus of arabic tweets for hate speech analysis," *arXiv preprint arXiv:2505.11969*, 2025.

[12] R. Daud, N. Basir, N. F. N. Mohd Rafei Heng, M. M. S. Meor Sepli, and M. Melinda, "Large language model versus human: A study of automatic data annotation in islamophobia dataset," *Available at SSRN 5045167*, 2024.

[13] A. S. Alzahrani, D. H. Alahmadi, N. M. Alharbi, and H. A. Almagrabi, "Blockchain-based crowdsourcing framework for machine learning ground truth," *IEEE Access*, 2025.

[14] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *NAACL-HLT*, 2019.

[15] C. Raffel *et al.*, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of Machine Learning Research*, 2020.

[16] W. Antoun, F. Baly, and H. Hajj, "Arabert: Transformer-based model for arabic language understanding," in *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools (OSACT4)*, 2020, pp. 9–15.

[17] A. Safaya, M. Abdullatif, and J. Ren, "Arabic-bert: Benchmarking pretrained language models for arabic," *arXiv preprint arXiv:2003.00104*, 2020.

[18] G. Inoue, N. Habash, D. Taji, H. Bouamor, and W. Zaghouani, "Camelbert: Transformer-based arabic language models pretrained on diverse corpora," *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pp. 91–103, 2021. [Online]. Available: https://aclanthology.org/2021.wanlp-1.10

[19] L. Xue *et al.*, "mt5: A massively multilingual pretrained text-to-text transformer," in *NAACL*, 2021.

[20] X. V. Lin *et al.*, "Few-shot learning with multilingual language models," *Transactions of the ACL*, 2022.

[21] OpenAI, "Gpt-4 technical report," 2023, https://openai.com/research/gpt-4.

[22] S. Khalifa *et al.*, "Evaluating chatgpt and gpt-4 for arabic tasks: A case study in sentiment and sarcasm," *arXiv preprint arXiv:2305.12345*, 2023.

[23] A. A. Alsuwaylimi and Z. S. Alenezi, "Leveraging transformers for detection of arabic cyberbullying on social media: Hybrid arabic transformers." *Computers, Materials & Continua*, vol. 83, no. 2, 2025.

[24] M. I. Khondaker, M. Mahmoud, M. Torres, and A. Nazarenko, "Gptaraeval: A comprehensive evaluation of chatgpt on arabic nlp tasks," *arXiv preprint arXiv:2308.08794*, 2023.

[25] S. Wang *et al.*, "Human-ai collaboration in data labeling: Benefits and pitfalls," in *Proceedings of ACL*, 2022.

[26] F. Senator, A. Lakhfif, I. Zenbout, H. Boutouta, and C. Mediani, "Leveraging chatgpt for enhancing arabic nlp: Application for semantic role labeling and cross-lingual annotation projection," *IEEE Access*, 2025.

[27] A. Bayazed, O. Torabah, R. AlSulami, D. Alahmadi, A. Babour, and K. Saeedi, "Sdct: multi-dialects corpus classification for saudi tweets," *methodology*, vol. 11, no. 11, 2020.

[28] S. Alzahrani, F. Alruwaili, D. Alahmadi, and K. Saeedi, "Tb-SAC: Topic-based and Sentiment Classification for Saudi Dialects Tweets," *International Journal of Computer Science and Network Security (IJCSNS)*, vol. 20, no. 9, pp. 41–49, 2020. [Online]. Available: http://ijcsns.org/07_book/html/202009/202009006.html

[29] A. Alsudais, "Comparing deep learning arabic sentiment analysis models using arsas dataset," *Journal of King Saud University-Computer and Information Sciences*, vol. 35, no. 1, pp. 74–81, 2023.

[30] S. Alhumoud and R. Alhajri, "Arabic sentiment analysis on twitter: A comparative study using deep learning and pretrained transformers," *Applied Sciences*, vol. 13, no. 5, p. 2741, 2023.

[31] Y. Liu, T. Han, S. Ma, J. Zhang, Y. Yang, J. Tian, H. He, A. Li, M. He, Z. Liu *et al.*, "Summary of chatgpt-related research and perspective towards the future of large language models," *Meta-radiology*, vol. 1, no. 2, p. 100017, 2023.

[32] A. Bahrini, M. Khamoshifar, H. Abbasimehr, R. J. Riggs, M. Esmaeili, R. M. Majdabadkohne, and M. Pasehvar, "Chatgpt: Applications, opportunities, and threats," in *2023 systems and information engineering design symposium (SIEDS)*. IEEE, 2023, pp. 274–279.

[33] R. Islam and O. M. Moushi, "Gpt-4o: The cutting-edge advancement in multimodal llm," *Authorea Preprints*, 2024.

[34] M. Hannani, A. Soudi, and K. Van Laerhoven, "Assessing the performance of chatgpt-4, fine-tuned bert and traditional ml models on moroccan arabic sentiment analysis," in *Proceedings of the 4th International Conference on Natural Language Processing for Digital Humanities*, 2024, pp. 489–498.

[35] M. Saranya and B. Amutha, "A comparative study on the evaluation of chatgpt and bert in the development of text classification systems," in *Generative Artificial Intelligence and Ethics: Standards, Guidelines, and Best Practices*. IGI Global, 2025, pp. 91–108.

[36] J. A. Lossio-Ventura, R. Weger, A. Y. Lee, E. P. Guinee, J. Chung, L. Atlas, E. Linos, and F. Pereira, "A comparison of chatgpt and fine-tuned open pre-trained transformers (opt) against widely used sentiment analysis tools: sentiment analysis of covid-19 survey data," *JMIR Mental Health*, vol. 11, p. e50150, 2024.

[37] CAMeL Lab, "bert-base-arabic-camelbert-da-sentiment," https://huggingface.co/CAMeL-Lab/bert-base-arabic-camelbert-da-sentiment, accessed: 2025-07-21.

Simulysis: A Method to Change Impact Analysis in Simulink Projects Based on WAVE-CIA

Hoang-Viet Tran*, Ta Van Thang, Cao Xuan Son, Do Trong Thu, Pham Ngoc Hung
University of Engineering and Technology, Vietnam National University,
Hanoi No. 144 Xuan Thuy Street, Dich Vong Ward, Cau Giay District, Hanoi, Vietnam

***Abstract*—MATLAB and Simulink, which have more than 5 million users and are installed at more than 100,000 businesses, universities, and government organizations[1], are widely used in numerous large-scale projects across various industries. These projects continually evolve in response to changes in business logic. However, managing the impact of these changes on Simulink projects presents several challenges to guarantee the quality of these projects. To address this, we propose a WAVE-CIA-based method named Simulysis for change impact analysis (CIA) in Simulink projects. The core idea behind Simulysis is to directly analyze Simulink project files and construct the project's corresponding call graph. By comparing the call graphs from the old and new project versions, Simulysis computes the change set. Subsequently, Simulysis applies the WAVE-CIA method to this change set and the call graph to identify the impact set. Additionally, Simulysis proposes a signal tracing method helping the system engineers to follow, check, and debug signals through the system. We have implemented Simulysis as a tool with the same name and conducted experiments using several open-source Simulink projects. The experiments demonstrate that Simulysis effectively performs the CIA process and retrieves the impact set, producing optimistic results and proving the practical applicability of Simulysis for real-world projects. Further discussions about Simulysis are provided in the paper.**

***Keywords*—Change impact analysis; WAVE-CIA; MATLAB/Simulink projects**

I. Introduction

In the development of a Simulink project, where the focus is on modeling, simulating, and analyzing dynamic systems, the evolution of requirements and functions is a common occurrence. Factors such as adding new features or accommodating customer preferences often drive changes. Similar to the software development life cycle, even minor adjustments in Simulink models can lead to unforeseen consequences and defects in the system. Consequently, change requests pose challenges for maintaining and managing Simulink projects. For developers, devising a strategy to identify the modules that require modification to accommodate changes is complex. Testers face the burden of testing numerous test cases after system modifications. Project managers grapple with cost allocation and time management related to change management. For this reason, understanding the relationships between altered Simulink components and their impact on other elements is crucial for Simulink projects and software projects in general. To address this, Change Impact Analysis (CIA) [1], [2], has been developed and applied. Numerous published papers discuss this solution [3], [1], [4], [5], [6], [2].

Researchers have indeed widely employed CIA across various systems, spanning different research fields and programming languages. In the domains of Business Process Management (BPM) and Service-Oriented Architecture (SOA), numerous published papers addressing CIA problems reveal that impact analysis solutions primarily fall into three categories: dependency analysis, traceability analysis, and history mining [4], [7]. First, dependency analysis is most frequently used. It encompasses various graph-based analyses, including control flow graph analysis [8], [9], message dependency graph [10], [11], trust dependency graph [12], directed hypergraph [13] and call graph analysis. Additionally, formal methods aid in change identification, formal modeling, and impact analysis [14], [15]. Quantitative analysis, which quantifies the depth of change using metrics, is essential for decision-making and problem-solving in service-based systems [16], [17], [18], [19], [20]. Lastly, execution trace analysis relies on information retrieved during program execution, either on-the-fly (online) or after program execution (offline) [21].

Following dependency analysis, traceability analysis plays a crucial role. It helps establish relationships between components and their impact on other elements, such as source code [22]. These relationships span multiple abstraction layers and are captured using traceability links. Lehnert's method enables automated or semi-automated impact analysis approaches for software, although it is not suitable for manual analysis [9]. Existing traceability can be categorized into vertical and horizontal traceability [23], [24]. Given the distributed nature of SOA, traceability analysis in SOA systems remains a challenging task.

History mining refers to the use of Mining Software Repositories (MSR) techniques to discover historical change dependencies [25], [21], [26]. This approach relies on analyzing the version histories of software to identify change patterns within source code repositories. By examining these patterns, the method can pinpoint processes or services that have frequently changed in tandem. It leverages change logs or version histories to predict the impact of future modifications. MSR methods have shown significant potential for impact analysis in software systems [27]. They have been applied across various aspects of software projects, including process model repositories [25], web service ecosystems [28], and process choreographies [29].

Change Impact Analysis (CIA) methods have been extensively implemented in various software languages. For instance, the CA Harvest Software Change Manager is a

[1]https://www.mathworks.com/content/dam/mathworks/fact-sheet/2023-company-factsheet-8-5x11-8282v23.pdf

comprehensive tool designed to track and manage software changes throughout the development lifecycle [30]. It supports C/C++ and several mainframe server languages like COBOL, PL/I, and Assembly, offering features that help organizations manage and understand the impact of their source code modifications. JRipples, an open-source tool tailored for Java developers, aids in comprehending programs during source code changes [31]. It manages steps for impact analysis and change propagation, highlights dependencies, and assists developers in understanding how changes in one part of the source code can affect another. JArchitect, although based on CIA, exclusively supports Java [32]. It's a source code analysis and visualization tool that helps developers enhance their source code. It offers a comprehensive set of features, including source code metrics, visualization of dependencies through directed graphs and dependency matrices, and the ability to compare source code snapshots. Lastly, several tools were implemented, including various CIA methods. SonarQube supports over 30 languages, frameworks, and Infrastructure as Code (IaC) platforms [33]. Checkmarx supports Java, C#, JavaScript, Python, Ruby, and PHP [34]. Ansible supports specific domain-based languages on YAML [35]. Jira provides an interface for collaboration and project management [36].

Regarding the CIA for Simulink projects, in 2017, Rapos et al. introduced a method to address the Change Impact Analysis (CIA) problem in Simulink projects [5]. This method, based on a set of MATLAB scripts, was implemented in a tool named SimPact. SimPact was validated as an impact predictor for the maintenance history of a large set of industrial models and their tests. However, SimPact can only operate within the Simulink Interactive Development Environment (IDE), meaning it cannot function independently of MATLAB. Other tools, such as Diffplug and BDT, support the impact analysis of Simulink projects using the signal tracing method [37], [6]. Despite these advancements, no published paper has yet addressed the CIA problem of Simulink projects that can operate as a standalone solution and utilize the WAVE-CIA method for performing the CIA.

This paper introduces Simulysis, a method based on WAVE-CIA [3], designed to identify and analyze the impact of changes between two versions of Simulink projects. Simulysis reads data from the project's *.mdl and *.slx files and builds the corresponding call graphs. It then compares thoroughly, identifying discrepancies and modifications between the two project graphs. Subsequently, Simulysis applies the WAVE-CIA to the changed Simulink components to identify the affected ones. Simulysis makes four main contributions. First, it introduces a technique for extracting information from Simulink files and classifying the changes in different project versions. Second, it demonstrates how to generate a call graph of the project, a feature not supported by the current version of MATLAB/Simulink. Third, Simulysis utilizes the WAVE-CIA method to analyze the impact of changes using the generated call graph. Finally, Simulysis proposes a method to trace a given signal in the provided Simulink project. This provides a crucial method for Simulink engineers to follow, check, and debug signals and signal-related bugs. We have implemented Simulysis as a tool and conducted preliminary experiments with promising results.

The structure of this paper is as follows. Section II introduces the essential background concepts. Section III delves into the details of the Simulysis methodology. Section III-B4 presents the details of the proposed signal tracing method. Section IV provides insights into the practical implementation of the Simulysis Tool. Initial experimental findings and subsequent discussions are presented in Section V. Section VI discusses the published papers related to Simulysis. The paper concludes in Section VII.

II. Backgrounds

In this section, we introduce background concepts that will be utilized in this paper.

A. Change Impact Analysis

Change Impact Analysis (CIA) [2] is a systematic approach commonly employed in software development, project management, and system engineering. Its primary purpose is to evaluate the consequences of modifications made to a system. CIA aids in making well-informed decisions, minimizing the introduction of errors, and optimizing resource utilization. While it cannot guarantee complete predictability or replace thorough testing in every instance, the benefits of CIA include improved system reliability, reduced change risks, and efficient resource allocation. In software projects, CIA benefits software developers, testers, project managers, system architects, and stakeholders involved in the change process. Overall, the CIA process contributes to better decision-making and enhances system quality.

B. WAVE-CIA

WAVE-CIA, introduced by Li et al. in 2013 [3], represents a novel Change Impact Analysis (CIA) approach based on the traditional call graph-based techniques. Drawing inspiration from the propagation of water waves, this method computes a set of impacted components based on a change set and a call graph. Much like the ripples formed when stones are thrown into the water, WAVE-CIA views the call graph as a water surface and the change set as the thrown stones. The central concept is that a component is more likely to be impacted if it has dependencies with altered components, and these modified components propagate the impact to others in the graph. By meticulously analyzing the call graph, WAVE-CIA identifies precise change ripples and adeptly handles multiple changes with interference. In comparison to traditional call graph-based CIA methods, WAVE-CIA achieves a more accurate impact set with fewer false positives, although it may occasionally miss a few false negatives.

In the context of Simulink, several terms and concepts differ from those found in conventional software development codebases. These terms have been mapped to suitable WAVE-CIA equivalents for implementation. The Simulink project comprises numerous Simulink files with extensions such as ".mdl" or ".slx". These files contain Simulink systems, which encompass various block types including Ports, Logic, References, Subsystems, and more, all interconnected by lines.

Definition 1 (Node or Vertex)**.** *Nodes represent systems in a Simulink project. Each node corresponds to a specific system, and the edges connecting the nodes represent the caller and*

callee relationships between these systems. This helps define how systems interact with each other in the project.

Definition 2 (Call Graph)**.** *A call graph is a directed graph G = (V, E), where V is a set of vertices representing Simulink components in the system, and $E \subseteq V \times V$ is a set of edges representing the call relationships between callers and callees. WAVE-CIA draws an analogy from water waves that ripple outward when stones are thrown into calm water. In the context of the call graph, which is akin to a water surface, Simulink components reachable by multiple altered components (meeting positions) are more likely to be impacted by the changes.*

Definition 3 (Change Set - CS)**.** *A change set refers to a group of Simulink files and systems within a Simulink project that have been modified or are marked for modification. This set is typically identified by comparing data between two versions of the same Simulink project.*

Definition 4 (Core Set)**.** *The core set is comprised of closely related Simulink components that are likely to be affected by multiple modifications in the change set. This set is generated by identifying Simulink components close to the change set within a call graph. These components, which are accessible by multiple modified components, are more susceptible to changes. The core set is determined by verifying if a component is influenced by more than one modified component, and it encompasses all the modified components in the change set. The likelihood of components in the core set being affected by multiple changes is high. Subsequently, the impact set is calculated using propagation analysis based on the core set.*

Definition 5 (Impact Set - IS)**.** *The impact set is made up of Simulink files and systems that could be affected by the Simulink components in the change set. It includes elements that may need to be reworked or revalidated. WAVE-CIA determines the impact set through the process of mining the call graph.*

C. Simulink

Simulink [38], a graphical extension to MATLAB [39] developed by MathWorks, is designed for modeling and simulating dynamic systems. It offers key advantages such as the capability to model nonlinear systems, which is not easily achievable with transfer functions, and the ability to include initial conditions for a more accurate depiction of system behavior. Simulink provides a graphical interface for users to build systems using block diagrams, incorporating various elements like transfer functions, adding nodes, virtual input/output devices, etc. Its integration with MATLAB ensures smooth data transfer between the two platforms. Due to its wide usage in system modeling, controller design, and simulation, Simulink is a potent tool in engineering and scientific applications.

Table I describes some common blocks used in Simulink, and Table II shows the data from the source code of Simulink blocks.

Additionally, we clarify the key terms used in Simulink modeling to ensure consistency throughout the paper. The terms *system*, *subsystem*, *component*, *block*, and *file* are fundamental to Simulink, but they have specific meanings and relationships that must be clearly understood.

Definition 6 (System)**.** *A system in Simulink is a complete model representing a specific dynamic behavior or a control logic. It encompasses all the elements necessary for simulating a particular functionality. A system can be a standalone model or can include multiple subsystems and components working together.*

Example: A vehicle dynamics model that includes the engine, transmission, and braking systems.

Definition 7 (Subsystem)**.** *A subsystem is a modular, hierarchical component of a system. It encapsulates a set of blocks that perform a specific function within the larger system. Subsystems help organize and manage complex models by breaking them into smaller, more manageable parts.*

Example: Within a vehicle dynamics system, the engine control unit can be a subsystem that handles all the engine-related computations and control logic.

Definition 8 (Component)**.** *A component in Simulink refers to any reusable, self-contained unit within a model. Components can be atomic or composite and can represent hardware or software elements. Components often refer to reusable libraries or pre-built blocks that can be integrated into various models.*

Example: A PID controller block can be a component used in various systems to manage control processes.

Definition 9 (Block)**.** *A block is the fundamental building element in Simulink models. Blocks represent mathematical operations, signal routing, or functional units. They can be combined to form systems and subsystems. Blocks can be simple (like a gain block) or complex (like a Simulink Function block).*

Example: A summation block that adds input signals together is a simple block, while a Stateflow chart is a more complex block.

Definition 10 (File)**.** *A file in the context of Simulink refers to the physical storage of models and components. Simulink models are typically saved with the .slx or .mdl file extensions. These files contain all the necessary information to define and simulate a model, including block diagrams, parameters, and configuration settings.*

Example: `vehicle_dynamics.slx` is a file that contains the complete Simulink model for vehicle dynamics.

D. Relationships in Simulink Project

To implement the CIA analysis of the Simulink projects, we have to identify the relationships between components and elements of Simulink projects.

1) Relationships between files: Within the context of file relationships, we establish connections based on the structure of the files in the project. This is essential for the implementation of WAVE-CIA, which depends on comprehending the dependencies among files. The various types of file relationships are illustrated in Table III.

TABLE I. COMMONLY USED SIMULINK BLOCKS

ID	Icon	Name	Description
1		Logical Operator	The Logical Operator block does the given logical operation on its inputs. If it is nonzero, the input value is true (1). Otherwise, it is false (0).
2		Relational Operator	The Relational Operator block does the given relational operation on the input. The value chosen for the Relational operator parameter determines if the block accepts one or two input signals.
3		Inport	Inport blocks in Simulink link signals from outside a system into the system.
4		Outport	Outport blocks in Simulink link signals from inside a system to a destination outside of the system. Outport blocks can connect signals originating from a subsystem to other parts of the model. Outport blocks can also supply external outputs at the top level of a model hierarchy.
5		Subsystem	A Subsystem block is a block that contains a subset of a model. We can have multiple subsystems in a model, and we can have subsystems inside other subsystems.
6		Constant	The Constant block of Simulink generates a real or complex constant value signal. We can use this block to provide a constant signal input.

Fig. 1 illustrates the relationships between various files. The file under examination in this figure is "Logic111". It is referenced within "Function12", which makes "Function12" the caller of "Logic111". Within the context of "Logic111", there are references to two other files: "Block1111" and "Logic112". This makes these two files the callees of "Logic111".

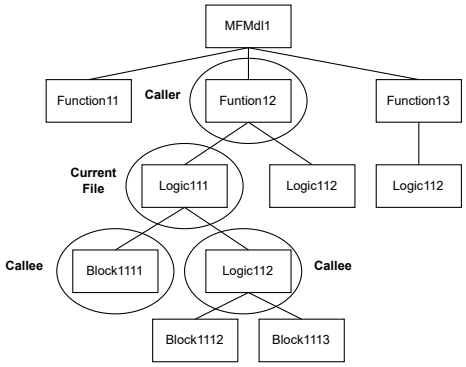


Fig. 1. Example of file relationships.

2) Relationships between systems: Within the context of Simulink systems relationships, we establish connections according to the structural organization of systems within a file. This definition is crucial for implementing WAVE-CIA.

Fig. 2 illustrates the relationships among systems. The Subsystem block "Function21" is currently under assessment for its relationships. It has two callers connected to it: the "Bit Set" block and the Inport block "In2_1". The output of "Function21" is directly linked to the Logical Operator block AND, which is therefore its callee.

E. Types of Change

In the process of analyzing two versions of a Simulink project, pinpointing the differences between them is crucial,

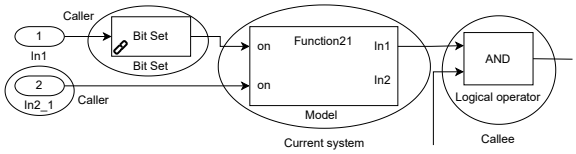


Fig. 2. Example of system relationships.

with a special emphasis on the modified components. The set of these modified elements is termed the change set (CS). Modifications are classified into three categories: addition, deletion, and change.

Simulysis employs the "Name" property of a Simulink system as the key to discern whether it has been added, deleted, or changed. This is a critical aspect of Simulysis, given that two systems within the same Simulink file cannot possess identical names. If a system name is present in both versions of the Simulink project, it is considered altered if there exists any discrepancy in its properties. Table V presents the various types of modifications in a Simulink project.

III. SIMULYSIS METHOD

A. Simulysis Overview

Simulysis is a method based on WAVE-CIA for performing CIA on Simulink projects. This method encompasses three primary phases in the analysis of a specified Simulink project. An overview of Simulysis is depicted in Fig. 3.

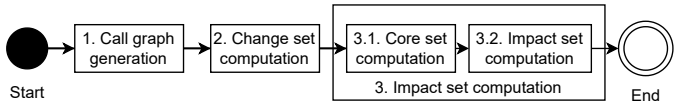


Fig. 3. An overview of simulysis.

TABLE II. SOURCE CODE OF SIMULINK BLOCKS

ID	Icon	Name	Source code
1		Logical Operator	Block source code: <Block BlockType="Logic" Name="Logical
Operator" SID="12"> <PortCounts in="2" out="1"/> <P Name="Position">[580, 152, 610, 183]</P> <P Name="ZOrder">12</P> <P Name="AllPortsSameDT">off</P> <P Name="OutDataTypeStr">boolean</P> </Block> Properties: - Position - ZOrder - AllPortsSameDT - OutDataTypeSt
2		Relational Operator	Block source code: <Block BlockType="RelationalOperator" Name="Relational
Operator" SID="16"> <PortCounts in="2" out="1"/> <P Name="Position">[270, 257, 300, 288]</P> <P Name="ZOrder">16</P> <P Name="Operator"><=</P> <P Name="InputSameDT">off</P> <P Name="OutDataTypeStr">boolean</P> <P Name="RndMeth">Simplest</P> </Block> Properties: - Position - ZOrder - Operator - InputSameDT - OutDataTypeStr - RndMeth
3		Inport	Block source code: <Block BlockType="Inport" Name="In1" SID="10"> <P Name="Position">[370, 153, 400, 167]</P> <P Name="ZOrder">10</P> </Block> Properties: - Position - ZOrder
4		Outport	Block source code: <Block BlockType="Outport" Name="Out1" SID="14"> <P Name="Position">[730, 158, 760, 172]</P> <P Name="ZOrder">14</P> </Block> Properties: - Position - ZOrder
5		Subsystem	Block source code: <Block BlockType="SubSystem" Name="Subsystem" SID="19"> <PortCounts in="1" out="1"/> <P Name="Position">[565, 229, 665, 271]</P> <P Name="ZOrder">19</P> <P Name="ContentPreviewEnabled">on</P> <System Ref="system_19"/> </Block> Properties: - Position - ZOrder - ContentPreviewEnabled - System Ref.
6		Constant	Block source code: <Block BlockType="Constant" Name="Constant" SID="3"> <P Name="Position">[380, 60, 410, 90]</P> <P Name="ZOrder">3</P> </Block> Properties: - Position - ZOrder

- Phase 1 (Call graph generation): Given a project, Simulysis obtains the project data by analyzing the project files. Next, it builds a call graph based on the structure and relationships between the Simulink systems (blocks) and files by using the callee map. A callee map is a set of Simulink components called by other components through the Outports. This phase is vital as it establishes the interconnections within the Simulink project and generates a call graph that will be utilized in subsequent phases.

TABLE III. FILE RELATIONSHIPS

ID	Relationship	Roles	Description
1	Parent	Caller	A file is deemed the parent and caller of another file when it contains references to that other file.
2	Children	Callee	A file is considered a child and callee of another file when it is referred to in the content of that other file.
3	No Relation		Two files are considered unrelated if they don't share a parent-child relationship, regardless of whether they have common parents or children.

TABLE IV. SYSTEM RELATIONSHIPS

ID	WAVE-CIA	Description
1	Caller	The system that triggers a call to another system is referred to as the "caller".
2	Callee	The system that receives a call from another system is referred to as the "callee".
3	No Relation	Two systems are considered unrelated if they are not directly connected.

TABLE V. TYPES OF CHANGE

ID	Change Type	Description
1	Addition	Simulink components that appeared in the new version but not the old version of the project are listed as Additions.
2	Deletion	Simulink components that are removed in the new version and existed in the old version of the project are listed as Deletions.
3	Change	Simulink components that still exist in both versions of the Simulink project but have different properties are listed as Changes.

- Phase 2 (Change set calculation): When provided with two call graphs corresponding to two versions of a Simulink project, Simulysis calculates the change set. This computed change set is then utilized in Phase 3 to determine the impact set.
- Phase 3 (Impact set computation): In this phase, the WAVE-CIA algorithm is implemented, using the change set from Phase 2 as an input. This process identifies the core set, relying on the call graph from Phase 1, and computes the impact set by implementing the WAVE-CIA algorithm. It's important to note that while Simulysis utilizes the WAVE-CIA technique for calculating the impact of changes, it's possible to employ any other CIA method that facilitates the computation of the impact set based on the call graph.

The subsequent sections delve into a comprehensive examination of each phase in the Simulysis method for Change Impact Analysis (CIA) of Simulink projects.

B. Call Graph Generation

There are many methods for the analysis of Simulink projects, such as manual navigation through the signals of the project [37], based on MATLAB-provided script [5], AI-based change distribution analysis [40], etc. By modeling a given Simulink project as a call graph, we have an efficient method to analyze, fix bugs, improve the model, trace the required signals, etc. In this paper, we employ the call graph for change impact analysis of Simulink projects for the two reasons below:

- Signal tracing: Simulink projects are designs of systems with many signals running across the model. By using the call graph to model the given project, signals can be seen as paths through the graph. For this reason, the signal tracing problem can be addressed by using the graph traversing problem and using such well-known algorithms as depth-first search, breadth-first search, etc. This greatly contributes to the system analysis and debugging of Simulink projects.
- Change impact analysis: By modeling a Simulink project as a call graph, the change impact analysis problem can be solved by applying such graph-based CIA methods as the WAVE-CIA method [3].

To create the call graph for a given Simulink project, Simulysis analyzes project files and generates the required call graph using the callee map. The creation of a call graph for a specified Simulink project involves the following three primary steps, as shown in Fig. 4.

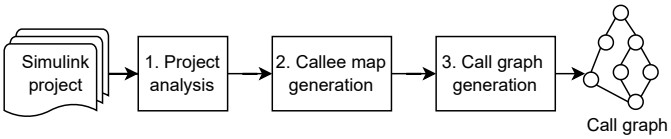


Fig. 4. The main steps of the call graph generation process.

- Step 1 (Project Analysis): Simulysis scans all the Simulink files in the specified project and extracts data about all existing systems, lines, and branches. Details of this step are presented in Section III-B1.
- Step 2 (Callee Map Generation): For every system within the project, Simulysis identifies other systems that are linked to it by employing a signal tracing algorithm. A system in Simulink serves as a computational unit that accepts input data via in-ports, processes this data, and generates output through the out-ports. Simulysis aims to locate the callees of the system, which are systems connected to the system's out-ports. The specifics of this step are illustrated in Algorithm 1.
- Step 3 (Call Graph Generation): Simulysis cycles through each system in the project, generating a new corresponding node in the call graph. Subsequently, Simulysis establishes connections to other nodes in the graph, utilizing the callers and callees of the system identified in Step 2. The specifics of this step are demonstrated in Algorithm 2.

1) Project analysis: For a specific Simulink project, Simulysis analyzes by scanning the list of Simulink files in the project and extracting a list of systems (blocks) and the lines that connect these systems.

The structure of a typical Simulink project and a Simulink file is depicted in Fig. 5 and Fig. 6. As a Simulink project comprises numerous files and folders, it is crucial to pinpoint the files where pertinent data is stored. Depending on the MATLAB version of the project, the primary information of a Simulink project is housed in various file types such as ".mdl"

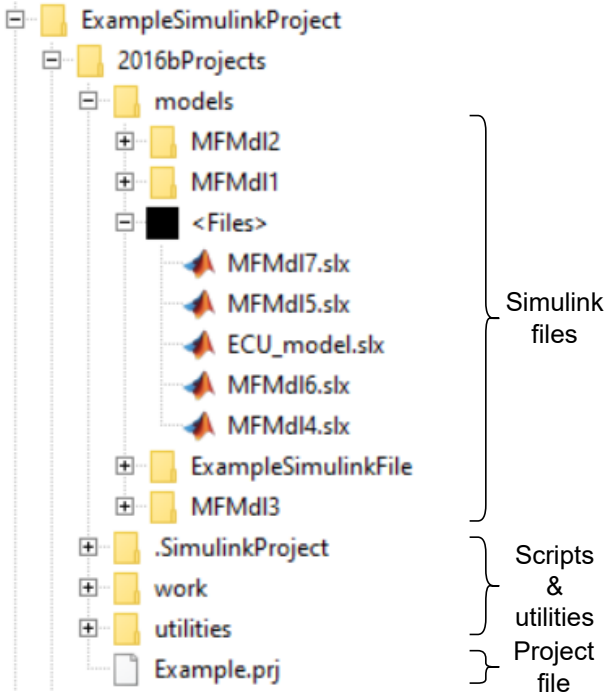


Fig. 5. Simulink project structure example.

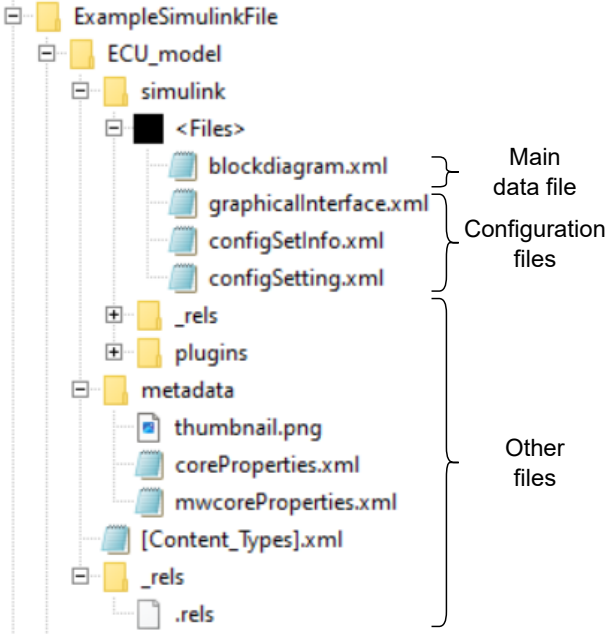


Fig. 6. Simulink file structure example.

and ".slx". Simulysis filters the project by verifying the validity of the model state and the file extension to locate files with the ".mdl" or ".slx" extension and other project information.

Before MATLAB version R2016b, Simulink files had the ".mdl" extension, and the data could be accessed directly. For newer versions, Simulink introduces a new ".slx" file extension, which is a zip file comprising multiple files and folders. To access the file, Simulysis alters the ".slx" file extension to ".zip" and then unzips the file. As only the main data file ("blockdiagram.xml") is needed to recreate a visualization and execute other functions, only this file is analyzed to retrieve system and line information.

Fig. 7 presents an instance of a "blockdiagram.xml" file within an ".slx" file. The file adheres to a standard XML format where the initial tags represent the file configuration and setting parameters. Simulysis utilizes only a few properties within these tags to construct the necessary call graph. The primary data is housed within the ¡System¿ tag, which encompasses the information of blocks (¡Block¿ tag) and lines (¡Line¿ tag). Leveraging the information from the blocks and lines, Simulysis assembles the required call graph for use in subsequent phases.

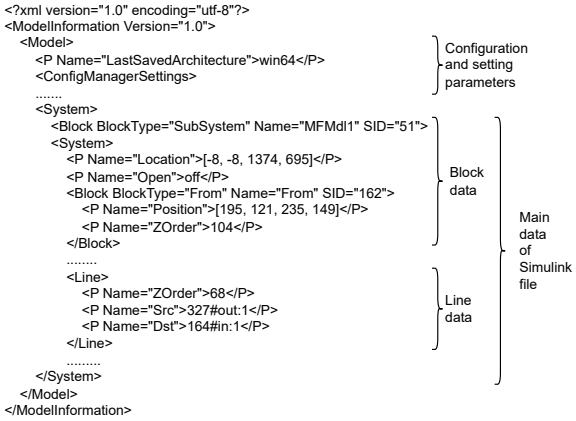


Fig. 7. Simulink file content example.

2) Callee map generation: The purpose of the callee map is to identify the subsystems (i.e., callees) associated with each system in the Simulink project. This is an important step in creating the call graph, which helps analyze call relationships between systems and determine the impact of changes in the project. Details of the callee map generation process are shown in Algorithm 1.

The algorithm starts by initializing the map ($systemCalleMap$) as an empty set ($\emptyset$) (line 2). It then iterates over each system, checking its type to determine the appropriate process (lines 3 - 30).

If a system is of type "$From$", it does not connect to other systems via lines. Instead, it directly jumps to a system of the "To" type with the same identifier, facilitating data transfer from the "$From$" system to the "To" system. Consequently, the algorithm identifies the corresponding "To" systems of the current "$From$" system and appends them as a caller-callee pair to the "$systemCalleeMap$" map (lines 4 - 6).

If a subsystem is then connected to another system, the algorithm will find the block corresponding to the port type and port number connected (lines 12 - 13). In Simulink, a system can connect to output port of type "$Output$", "$LConn$" and "$RConn$" (in which "$LConn$" and "$RConn$" represents left and right side physical connection port - a special port type in Simulink). The corresponding block that links a signal from inside the subsystem to outside the subsystem is of type "$Outport$", "$PMIOPort$ with "$Side$" attribute equals "$Left$", "$PMIOPort$ with "$Side$" attribute equals "$Right$".

Algorithm 1 Generate the Callee Map

Input: $systemList$: List of systems of the project
$lineList$: List of lines of the project
Output: The required callee map

```
1  begin
2      systemCalleeMap ← ∅
3      foreach system in systemList do
4          if system type is "From" then
5              calleeSystem ← findGoto(systemList, system)
6              systemCalleeMap.append(system, calleeSystem)
7          else
8              lines ← findLines(lineList, system)
9              foreach line in lines do
10                 callerSystem ← system
11                 calleeSystems ← ∅
12                 if system type is "SubSystem" then
13                     callerSystem ← getOutputPortBlock(system, line)
14                 if line has branches then
15                     foreach branch in line.getBranches() do
16                         calleeSystem ← findSystem(systemList, branch)
17                         if calleeSystem type is "SubSystem" then
18                             calleeSystem ← getInputPortBlock(calleeSystem, line)
19                         calleeSystems ← calleeSystems ∪ calleeSystem
20                     end
21                 else
22                     calleeSystem ← findSystem(systemList, line)
23                     if calleeSystem type is "SubSystem" then
24                         calleeSystem ← getInputPortBlock(calleeSystem, line)
25                     calleeSystems ← calleeSystems ∪ calleeSystem
26                 end
27                 systemCalleeMap.append(callerSystem, calleeSystems)
28             end
29         end
30     end
31     return systemCalleeMap
32 end
```

After finding out the corresponding output port blocks, the algorithm assigned it as the caller (line 13)

If the system type is not "$From$", the algorithm identifies lines connected to it. In Simulink, a line block contains some branch properties, which are lines connecting to other blocks. Therefore, for each line, the algorithm checks if the line contains any branches. If so, the algorithm iterates through the branches to find the system that is connected to them and adds them to the callee list ($calleeSystem$) (lines 14 - 20). If not, the algorithm finds the system connecting to the line and adds it to the callee list ($calleeSystem$) (lines 21 - 26).

If a system is connected to a system of type "$SubSystem$" through the subsystem's input ports (lines 17 - 18 and 23 - 24), the algorithm will find the port block inside the subsystem correspond to port type and port number connected. In Simulink, a system can connect to an input port of type "$Input$", "$LConn$" and "$RConn$". The correspond block that links signal from outside the subsystem to the inside are of type "$Inport$", "$PMIOPort$ with "$Side$" attribute equals "$Left$", "$PMIOPort$" with "$Side$" attribute equals "$Right$". After finding all corresponding input port blocks, the algorithm then and adds them to the "$calleeSystems$" array (line 19 and 25).

After finding out the caller and the callee systems, the algorithm adds the caller system ($callerSystem$) and its corresponding callee list ($calleeSystems$) to the $systemCalleeMap$ map (line 27). Once all systems in $systemList$ have been processed, the algorithm returns $systemCalleeMap$.

Time Complexity Analysis

Theorem III.1. *Let n be the total number of systems in $systemList$, m be the maximum number of lines in the $lineList$ and k be the maximum number of lines in the $line.getBranches()$. The time complexity of Algorithm 1 is O($n * m * k$).*

Proof: From Algorithm 1, we see that the $foreach$ loop at lines 3 - 30 iterates n times, resulting in time complexity of this loop is O(n). Lines 4 - 6 check the type of each system. If the $systemType$ is "$From$", its find the corresponding "To" system. This check has a time complexity O(1) for each system. Line 8 finds $lines$ connected to the system, which has time complexity O(1) for each system. Line 9 - 28 iterate through each line connected to the system: Line 14 - 20 handle lines with branches, iterating through each branch and finding the connected systems. This has a time complexity of O(k) in the worst case, where k is the maximum number of branches for any $line$ in $lines$. Line 21 - 26 handle lines without branches, finding the connected system. This has a time complexity of O(1) for each line. Since we need to consider all lines of the system, the time complexity for processing lines is O($m * k$). Combining these steps, the total time complexity for processing each system and its connected lines is O($m * k$) in the worst case. For this reason, iterating over all n systems results in an overall time complexity of O($n * m * k$) of Algorithm 1. ■

3) Call graph generation: Once the callee map is prepared, Simulysis proceeds to generate the call graph, which will be utilized later in the CIA process.

The generation of this call graph for a given Simulink project is detailed in Algorithm 2.

The algorithm takes as input the list of systems from Step 1 ($systemList$) and the callee map from Step 2 ($systemCalleeMap$). After processing, the algorithm outputs the desired call graph. The process begins by initializing the call graph as an empty set ($\emptyset$). It's important to note that Simulink already generates a unique identifier, known as SID. For this reason, Simulysis leverages this SID value as the identifier for each node in the call graph. The algorithm iterates over each system ($system$) in the system list ($systemList$), retrieving the SID of the system (line 4). It then either retrieves an existing node or creates a new one in the call graph ($callGraph$) that corresponds to the system ($system$) (line 5). Next, the algorithm fetches the callee list ($calleeList$) of the system ($system$) from the callee map ($systemCalleeMap$) (line 6). It then cycles through the callee list ($calleeList$), identifies the node corresponding to the callee system node in the call graph, and establishes an edge between the callee and caller (lines 7 - 11). This entire process is repeated for each system (lines 3 - 12). Finally, the algorithm returns the call graph (line 13).

Time Complexity Analysis

Algorithm 2 Generate the Corresponding Call Graph for a Simulink Project

Input: $systemList$: List of systems in the project
$systemCalleeMap$: A map of callees by system
Output: $callGraph$: The corresponding call graph of the project

```
1  begin
2      callGraph ← ∅
3      foreach system in systemList do
4          systemSID ← getSystemSID(system)
5          node ← getOrCreateNode(callGraph, systemSID)
6          calleeList ← getCallees(systemCalleeMap, system)
7          foreach callee in calleeList do
8              calleeSystemSID ← getSystemSID(callee)
9              calleeNode ← getOrCreateNode(callGraph, calleeSystemSID)
10             createEdge(callGraph, node, calleeNode)
11         end
12     end
13     return callGraph
14 end
```

Theorem III.2. *Let n be the total number of systems in $systemList$, and m be the maximum number of callee systems for any system in $systemList$. The time complexity of Algorithm 2 is O($m * n$).*

Proof: From Algorithm 2, we see that the $foreach$ loop at lines 3 - 12 iterates n times, resulting in time complexity of this loop is O(n). Given a specific system ($system$) in the loop, other actions in this loop at lines 4 - 6 are getting the system id ($systemSID$), its corresponding node ($node$), and its list of callee systems ($calleeList$). These are simple actions that retrieve data from a data store such as a database or file, resulting in a time complexity of O(1). Let m be the maximum number of callee systems in all $calleeList$, the second loop (line 7 - 11) will have time complexity of O(m). As a result, the overall time complexity of this algorithm is O($m * n$). ■

4) Signal tracing method: In large Simulink projects, when facing a bug, especially a signal-related bug, system engineers must be able to fix the bug by checking the signal to find the bug. We call this the signal tracing problem. This is a complex task as the signal can go from the inport through multiple levels of the system model to the outport of the model. Using the corresponding call graph, the signal tracing problem can be modeled as the path-finding problem inside a directed graph.

It is important to identify the relationships between objects in Simulink models to implement the Depth-First Search algorithm (DFS). To reduce the complexity of the algorithm, Simulysis proposes identifying relationships between objects, including systems, lines, and branches. There are six types of relationships: System-Line, Line-Branch, Branch-Branch, Branch-System, From-Goto, and Parent-Child.

System-Line Relationship The System-Line relationship is the most common in the Simulink model, defined through the object's ID attribute. Fig. 8 illustrates an example of information about systems and one line in an ".slx" file. It can be seen that objects are connected through ID values; each block has a unique ID value that does not duplicate any other block within the same file. The line will have two child tags <P> with the attributes "Name=Dst" and "Name=Src". The value of the "Name=Dst" tag will match the ID value of a system in the file, indicating that it is the destination system. The value of the "Name=Src" tag will indicate the source system.

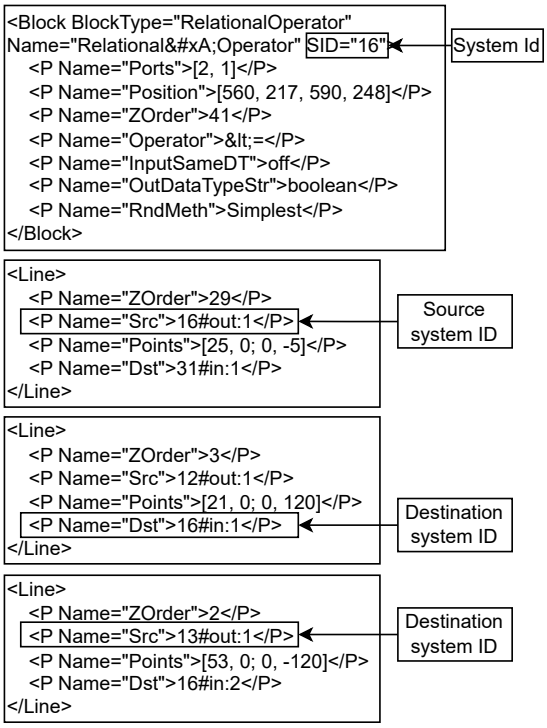


Fig. 8. System-line relationship.

System-Branch Relationship

Fig. 9 illustrates an example of the System-Branch relationship. Similar to the System-Line relationship, the System-Branch relationship is determined by the ID value. The value of the child tag "Name=Dst" will also indicate the destination system connected by the branch. However, unlike a line, a branch will not have a child tag <P> with the attribute "Name=Src".

Line-Branch Relationship

A branch is defined as a child object of a line in Simulink in the model. A connection line can have multiple branch lines. Fig. 10 describes the information about a line in the ".slx" file. The <Line> tag contains two child tags ¡Branch¿ that represent the line with two corresponding branch lines.

Branch-Branch Relationship

Similar to the Line-Branch relationship, the Branch-Branch relationship is formed when a branch contains child tags that are other branches, as shown in Fig. 11.

Goto-From Relationship

Goto and From are two special types of blocks, where a Goto block can transmit a signal to a corresponding From block and vice versa. The notable point is that these two

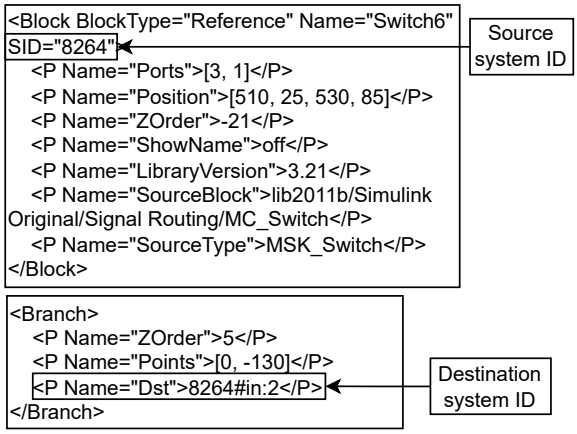


Fig. 9. System-Branch Relationship.

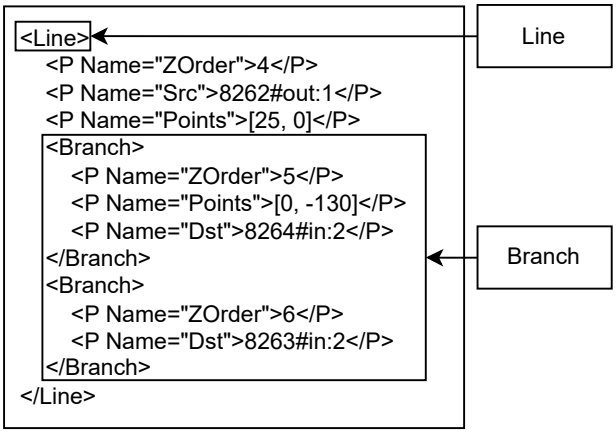


Fig. 10. Line-branch relationship.

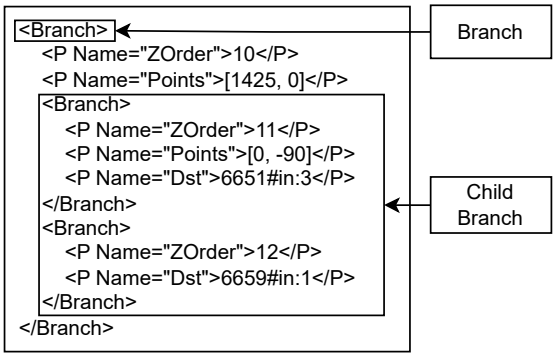


Fig. 11. Branch-branch relationship.

blocks do not connect with each other through a line. A Goto block can connect with multiple From blocks located at different positions in the model. The Goto-From relationship is determined through the GotoTag parameter. Two blocks with the same GotoTag value are considered to be connected. As shown in Fig. 12, the GotoTag value of both blocks is "xclcn", so they are connected.

Parent-Child Relationship

A system is called a parent if it contains a subsystem; the

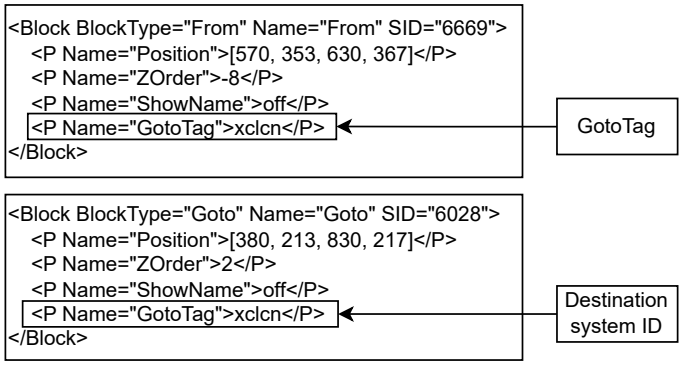


Fig. 12. Goto-from relationship.

objects within the subsystem will be children of that block. In the structure of a MATLAB Simulink model, Simulysis identifies three types of systems that contain subsystems: "Subsystem", "ModelReference", and "Reference". The "ModelReference" and "Reference" blocks have similar structures; the files containing the subsystems of these two types of blocks will be in different files within the project. The "Subsystem" block has a simpler structure, with its subsystem contents residing in the same file as the current model. Fig. 13 describes an example of a line being the input to a subsystem.

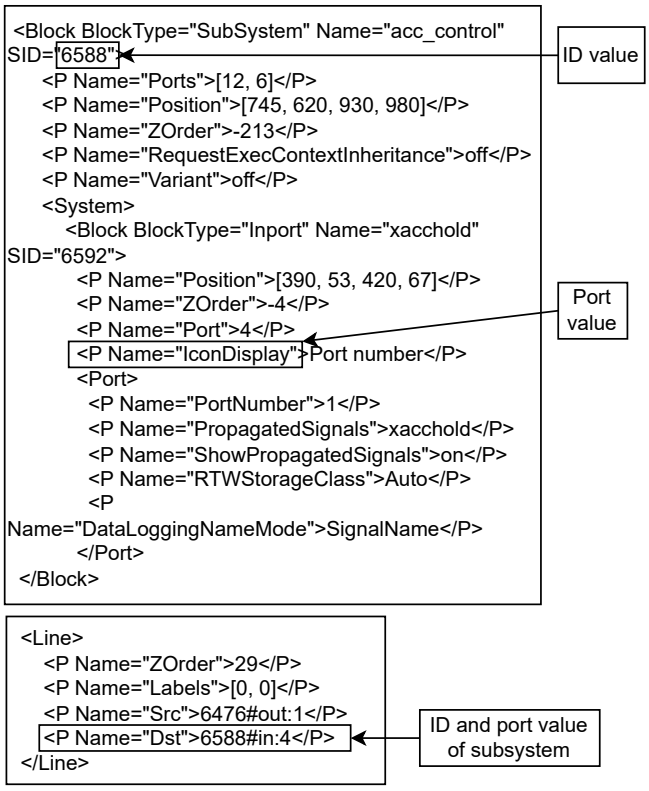


Fig. 13. Parent-child relationship

In this figure, the subsystem will be connected to the parent system through the Port value. A line/branch line will be an input to a subsystem if it contains a child tag <P> with the attribute "Name=Dst" having an ID value matching the parent system's ID value and the destination Port value matching an

input Port value of the parent system. Similarly, a line will be an output of the subsystem if it contains a child tag <P> with the attribute "Name=Src" having an ID value matching the parent system's ID value and the source Port value matching an output Port value of the parent system.

Signal Tracing Algorithm

Simulysis uses a DFS algorithm to traverse the call graph with the support of a stack of nodes. We maintain the three global parameters to support the DFS traversing algorithm. First, $pathList$ is the list of all paths from the $startNode$ to $endNode$. Second, $path$ is a temporary variable for storing the path being processed. Third, $stack$ is a stack data structure to store nodes of the call graph being processed. The algorithm accepts $startNode$ and $endNode$, which are the start and end nodes of the signal being checked. When the algorithm finishes running, $pathList$ stores the list of all paths from $startNode$ to $endNode$. Details of the algorithm are shown in Algorithm 3.

Algorithm 3 Signal Tracing

Input: $startNode \in callGraph$: the start node of the $path$ being checked.
$endNode \in callGraph$: the end node of $path$ being checked.
Output: The list of paths is stored in $pathList$

```
1  begin
2      pathList, path, stack ← ∅
3      stack.push(startNode)
4      while stack is not empty do
5          curNode ← stack.pop()
6          path ← path ∪ {curNode}
7          if curNode is endNode then
8              newPath ← clone_path(path);
9              pathList ← pathList ∪ {newPath});
10             if stack is not empty then
11                 curNode ← stack.pop()
12                 path ← remove_visited_nodes(path, curNode)
13                 call Algorithm 3(curNode, endNode);
14             end
15         else
16             childList ← get_child_nodes(curNode)
17             foreach child in childList do
18                 stack.push(child)
19             end
20             if stack is not empty then
21                 curNode ← stack.pop()
22                 call Algorithm 3(curNode, endNode);
23             end
24         end
25     end
26 end
```

The algorithm starts by initializing the path list ($pathList$), temporary path ($path$), and the stack data structure ($stack$) as empty sets ($\emptyset$) (line 2). Then, the algorithm push $startNode$ to the $stack$ (line 3). While the $stack$ is not empty, the algorithm does the following smaller tasks. First, the algorithm retrieves the top node from the stack ($stack$) and adds it to the temporary path ($path$) (lines 5 - 6). In case the current node ($curNode$) is the end node ($endNode$), we have found one path from the start node ($startNode$) to the end node ($endNode$). The algorithm adds the cloned path ($newPath$) of $path$ to the path list ($pathList$) (lines 8 - 9). We clone the path to ensure that later changes does not affect the cloned path. Next, the algorithm finds the next path from the start node ($startNode$) to the end node ($endNode$). If the stack ($stack$) is not empty, the algorithm retrieves the current node ($curNode$) from the top node of the stack (line 11). After getting the previous path, currently, $path$ still contains nodes from the previous path. We remove all unnecessary nodes from $path$ (line 12). These nodes are all nodes from the end of $path$ to the sibling nodes of $curNode$ that previously belonged to $path$. Then, the algorithm recursively calls itself to find paths from the current node ($curNode$) to the end node ($endNode$) (line 13).

In case the current node ($curNode$) is not the end node ($endNode$), the algorithm retrieves all next nodes (we call them child nodes) ($childList$) in the directed call graph (line 16) and pushes them to the stack ($stack$) (lines 17 - 19). Then, if the stack ($stack$) is not empty, the algorithm retrieves the top node ($curNode$) and recursively calls itself to find path from the current node ($curNode$) to the end node ($endNode$) (lines 20 - 23).

The time complexity of Algorithm 3 is $O(V, E)$, where V and E is the sets of nodes and edges in the call graph [41].

C. Change Set Computation

Simulysis generates call graphs corresponding to two versions of a Simulink project, as outlined in Algorithm 2. For impact analysis using the WAVE-CIA method, it's necessary for Simulysis to identify the change set between these two call graph versions. The process relies on the use of the "Name" and "SID" properties of a Simulink block to determine its presence across both graph versions. If a block is found in both versions, Simulysis examines if any of its properties have changed. If a change is detected, the block is marked as changed. The specifics of this process are detailed in Algorithm 4.

Algorithm 4 Compute the Change Set

Input: $callGraphBefore$, $callGraphAfter$
Output: $changeSet$

```
1  begin
2      changeSet, changedList, addedList, deletedList ← ∅
3      foreach node in callGraphAfter do
4          if ¬ NodeExistsIn(node, callGraphBefore) then
5              AddToList(node, addedList)
6          end
7          if NodeExistsIn(node, callGraphBefore) and
             PropertyChanged(node, callGraphBefore,
             callGraphAfter) then
8              AddToList(node, changedList)
9          end
10     end
11     foreach node in callGraphBefore do
12         if ¬ NodeExistsIn(node, callGraphAfter) then
13             AddToList(node, deletedList)
14         end
15     end
16     changeSet ← toSet(addedList, deletedList,
         changedList)
17     return changeSet
18 end
```

Algorithm 4 takes in two call graphs ($callGraphBefore$, $callGraphAfter$) corresponding to two versions of a given Simulink project. The output is the required change set ($changeSet$). The algorithm begins by initializing the change set ($changeSet$) and three temporary lists: $changedList$, $addedList$, and $deletedList$ as empty set ($\emptyset$). For each $node$ in $callGraphAfter$, the algorithm checks if the node also exists in $callGraphBefore$. If it does not, the node is considered as added and is added to the added node list ($addedList$) (lines 4 - 6). If the node exists in $callGraphBefore$ and any of its properties have changed in the two graphs (detected by the changes in the text of properties), the node is added to the change list ($changedList$) (lines 7 - 9). The algorithm then checks for any node that exists in $callGraphBefore$ but not in $callGraphAfter$. Such a node is considered as deleted and is added to the deleted node list ($deletedList$) (lines 11 - 15). Finally, the algorithm adds the three lists: $addedList$, $deletedList$, and $changedList$ to the $changeSet$ (line 16) and returns the result.

Time Complexity Analysis

Theorem III.3. *Let n be the maximum number of nodes in the call graphs for both the previous and updated project versions. The time complexity of Algorithm 4 is O(n).*

Proof: Let m be the number of nodes in the call graph corresponding to the new project version ($callGraphAfter$). From Algorithm 4, it is evident that the $foreach$ loop at lines 3 - 10 has a time complexity of O(m). In this loop, thanks to the utilization of the hash table data structure, the action that verifies if a node exists in a call graph has a time complexity of O(1) (line 4 and line 7). The functions $PropertyChanged$ and $AddToList$ are straightforward functions with a time complexity of O(1), resulting in the time complexity of this loop being O(m). Let k be the number of nodes in the call graph corresponding to the old project version ($callGraphBefore$). Following the same reasoning as above, the loop at lines 11 - 15 would have a time complexity of O(k). The $toSet$ function at line 16 is a simple function with a complexity of O(1). From the above analysis, let $n = max(m, k)$, the overall time complexity of the algorithm is O($max(m, k)$)≡O(n). ■

D. Impact Set Computation

Upon obtaining the change set from Phase 2, Simulysis proceeds to compute the impact set. This set includes Simulink files and systems that are influenced by the Simulink components present in the change set. Given a call graph and a change set, which is a subset of the call graph, Simulysis follows a two-step process to calculate the impact set. The first step involves the computation of the core set, which includes the computation of the neighbor set as detailed in Section III-D1. The second step is the computation of the impact set itself.

1) Neighbor Set Computation: In WAVE-CIA, a core set comprises Simulink systems that are potentially influenced by several systems within the change set. The computation of the core set utilizes the concept of a neighbor set associated with a given component. A neighbor set of a specific component, with a depth of n, includes other components in the call graph that can be reached via a maximum of n edges.

The parameter "depth" plays a crucial role in the zoning of the Coreset. The depth parameter significantly impacts the accuracy and efficiency of the Coreset. Depth determines the level at which data points are considered and clustered within the Coreset. A higher depth value implies a finer granularity, resulting in more detailed and potentially more accurate coresets, whereas a lower depth value leads to a more generalized and coarser representation of the data. To identify the optimal depth for a specific problem, empirical experimentation is essential. The appropriate depth varies depending on the nature of the dataset and the specific requirements of the analysis. By conducting experiments with varying depth values, we can evaluate the performance and accuracy of the resulting Coreset, thereby determining the most suitable depth parameter for their particular application.

Simulysis employs a recursive algorithm to identify the neighbors of a specific system node within the call graph. The procedure for determining the neighbor set is detailed in Algorithm 5.

Algorithm 5 Compute the Neighbor Set

Input: $systemNode$: a given system in a call graph
$depth$: The maximum edges between two Simulink to consider neighbors
$callGraph$: The call graph being analyzed
Output: $neighborSet$: The required neighbor set

```
1  begin
2      systemNode.isVisited ← true
3      neighborSet ← ∅
4      callers ← findCallers(systemNode, callGraph)
5      callees ← findCallees(systemNode, callGraph)
6      neighborSet ← neighborSet ∪ callers ∪ callees
7      if depth ≤ 1 then
8          return neighborSet
9      end
10     foreach neighborNode ∈ neighborSet and
         neighborNode.isVisited = false do
11         subNeighborSet ← call Algorithm 5(neighborNode,
             depth − 1, callGraph)
12         neighborSet ← neigborSet ∪ subNeighborSet
13     end
14 end
```

The algorithm initiates by marking the $systemNode$ as visited (line 2) and setting the $neighborSet$ as an empty set ($\emptyset$) (line 3). It then identifies the direct caller ($callers$) and direct callee ($callees$) lists of the current node, and appends these lists to the neighbor set (lines 4 - 5). This step locates all neighbor nodes that are connected to the current $systemNode$ via a single edge. The recursion terminates when $depth \leq 1$ (lines 7 - 9). Subsequently, for each unvisited $neighborNode$ in the $neighborSet$, the algorithm recursively calls itself to find the neighbor set ($subNeighborSet$) of the $neighborNode$ and adds it to the $neighborSet$ (lines 10 - 13). In this step, the algorithm locates all direct neighbors of the $neighborNode$, hence the parameter $depth$ must be $depth-1$. Upon termination of the algorithm, we obtain the neighbor set of the given $systemNode$, with a $depth$ number equivalent to the number of edges from the $systemNode$.

2) Time Complexity Analysis:

Theorem III.4. *Let n be the total number of nodes in the call graph, d be the maximum degree of any node in the call graph, and k be the specified depth. The time complexity of Algorithm 5 is O($n * d^k$).*

Proof: From Algorithm 5, we see that lines 4 - 5 finds direct callers and callees of $systemNode$ and adds them to $neighborSet$ which involves checking connections and has a time complexity of O(d) since d is the maximum degree of any node. Lines 7 -9 checks if $depth$ is less than or equal to 1 and, if so, returns $neighborSet$, which has a time complexity of O(1). Line 10 - 13 handle the recursive exploration of neighbors. For each unvisited neighbor in $neighborSet$, the algorithm recursively calls itself with $depth-1$. This recursion explores up to k levels, and at each level, it may process up to d neighbors. The time complexity for exploring neighbors at each level is O(d), and since the recursion depth is O(d^k). Given that there are n nodes in the call graph and in the worst case, each node needs to process its neighbors, the overall time complexity of the algorithm is O($n * d^k$). As a result, the overall time complexity of this algorithm is O($n * d^k$). ■

3) Core Set Computation: The computation of the core set is a crucial step in the WAVE-CIA method. This computation depends on the change set, and its outcome is subsequently utilized to calculate the impact set. The specifics of the core set computation process can be found in Algorithm 6.

Algorithm 6 Compute Core Set

Input: $callGraph$: The call graph of the Simulink project
$changeSet$: A set of components that have been changed
$depth$: An integer specifying the neighbor set exploration depth
Output: $coreSet$: the required core set

```
1  begin
2      coreSet ← ∅
3      foreach node ∈ callGraph do
4          if node ∉ changeSet then
5              neighborSet ← call Algorithm 5(node, depth, callGraph)
6              if —neighborSet ∩ changeSet— > 1 then
7                  coreSet ← coreSet ∪ {node}
8              end
9          end
10     end
11     coreSet ← coreSet ∪ changeSet
12     return coreSet
13 end
```

The algorithm takes in three inputs: the call graph ($callGraph$), the change set ($changeSet$), and an integer ($depth$) that determines the extent of the neighbor set computation. The output is the desired core set ($coreSet$). The algorithm starts by initializing the core set as an empty set ($\emptyset$) (line 2). It then iterates over each node ($node$) in the call graph ($callGraph$) that is not in the change set ($changeSet$), and invokes Algorithm 5 to compute the neighbor set ($neighborSet$) of the node (line 5). If the $neighborSet$ and $changeSet$ share more than one node, the node is added to the $coreSet$ (lines 6 - 8). After all nodes in the $callGraph$ have been checked, he algorithm incorporates the $changeSet$ into the $coreSet$, as the core set is a superset of the change set (line 11). The algorithm concludes by returning the $coreSet$ (line 12).

Time Complexity Analysis

Theorem III.5. *Let n be the total number of nodes in the call graph, d be the maximum degree of any node in the call graph, and k be the specified depth. The time complexity of Algorithm 5 is $\mathcal{O}(n \times d^k)$.*

Proof: From Algorithm 6, we see that line 2 initializes the core set with a time complexity of $\mathcal{O}(1)$. Lines 3 - 10 iterate through each node in the call graph that is not in the change set. For each node, line 5 calls Algorithm 5 to compute the neighbor set, which has a time complexity of $\mathcal{O}(d^k)$ since Algorithm 5 explores up to k levels of neighbors. Line 6 checks the intersection of the neighbor set and the change set, which involves set operations with a time complexity of $\mathcal{O}(d)$. Line 7 adds the node to the core set if the intersection condition is met, with a time complexity of $\mathcal{O}(1)$. Line 11 merges the change set into the core set, which has a time complexity of $\mathcal{O}(n)$ in the worst case. Given that there are n nodes in the call graph and each node's neighbor set computation involves $\mathcal{O}(d^k)$ operations, the overall time complexity of the algorithm is $\mathcal{O}(n \times d^k)$. ■

4) Impact set calculation: Although the core set is expected to encompass many components affected by the change sets, other impacted components may not be included in the core set. Consequently, it becomes necessary to broaden the change ripple to account for additional impacts that might have been overlooked, or in simpler terms, to enlarge the core. The fundamental concept of this procedure is that any node not already in the impact set will be added if both its caller and callee sets share nodes with the existing impact set. The specifics of this procedure can be found in Algorithm 7.

Algorithm 7 Compute the Impact Set

Input: $callGraph$: The call graph of the Simulink project
$coreSet$: A core set
Output: $impactSet$: the required impact set

```
1  begin
2      impactSet ← coreSet
3      isStable ← false
4      while not isStable do
5          isStable ← true
6          foreach node ∈ callGraph do
7              if node ∉ impactSet then
8                  callerSet ← findCallers(node, callGraph)
9                  calleeSet ← findCallees(node, callGraph)
10                 if —callerSet ∩ impactSet—>0 and —calleeSet ∩ impactSet—>0 then
11                     impactSet ← impactSet ∪ {node}
12                     isStable ← false
13                 end
14             end
15         end
16     end
17     return impactSet
18 end
```

Algorithm 7 takes in two inputs: the call graph ($callGraph$) and the core set ($coreSet$). It outputs the desired impact set

($impactSet$). The algorithm begins by initializing the impact set ($impactSet$) with the core set ($coreSet$) (line 2). It then continuously checks each node in the call graph that is not in the impact set to determine if it has at least one caller and one callee within the impact set. If it does, the node is considered to be impacted by the change set (lines 6 - 15). This process is repeated until the impact set reaches a state of stability, where no additional nodes are added during the checking process. The stability of the impact set is monitored using a boolean flag, $isStable$ (line 3), which is initially set to true $true$ (line 5) at the start of the call graph checking loop (lines 6 - 15). This flag is reset to $false$ when a new node is added to the impact set (line 12). If the flag remains $false$ after the loop, the loop is rerun. If not, the algorithm terminates the loop and returns the impact set ($impactSet$) (line 17).

Time Complexity Analysis

Theorem III.6. *Let n be the total number of nodes in the call graph, and d be the maximum degree of any node in the call graph. The time complexity of Algorithm 6 is $\mathcal{O}(n \times d)$.*

Proof: From Algorithm 7, we see that line 2 initializes the impact set with a time complexity of $\mathcal{O}(1)$. The outer loop (lines 4 - 16) repeats until no changes occur, and within each iteration, the inner loop (lines 6 - 15) iterates through each node in the call graph. Line 10 checks the intersections of the caller set and callee set with the impact set, each of which involves set operations with a time complexity of $\mathcal{O}(d)$. Lines 11 - 12 add the node to the impact set if the conditions are met and set the changed flag, both of which have a time complexity of $\mathcal{O}(1)$. Assuming the algorithm stabilizes after a constant number of iterations, the time complexity for each iteration through the call graph is $\mathcal{O}(n \times d)$. Therefore, the overall time complexity of the algorithm is $\mathcal{O}(n \times d)$. ■

IV. Implementation

We've developed a tool named Simulysis[2], utilizing the MVC framework and C#.NET within the Microsoft Visual Studio IDE. The architecture of Simulysis, depicted in Fig. 14, comprises three primary components.

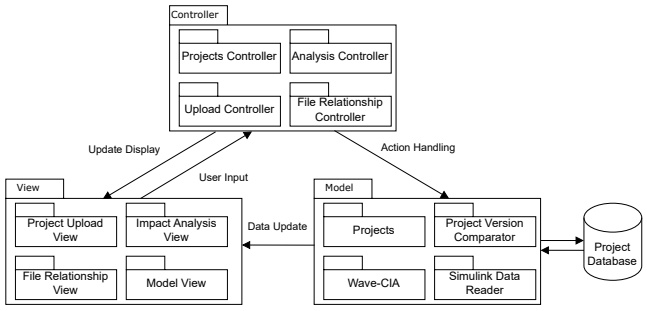


Fig. 14. The Simulysis architecture.

- Model: The Model component, encompassing four sub-components, is tasked with managing project data. The Projects sub-component oversees the information of Simulink projects. The Simulink Data Reader extracts and handles crucial data from Simulink projects. The Project Version Comparator sub-component performs comparisons between different versions of Simulink projects. Lastly, the WAVE-CIA sub-component carries out the change impact analysis.
- Controller: The Controller component, composed of four sub-components, acts as a bridge, overseeing user interactions and the flow of the application. The four sub-components include the Upload Controller, Project Controller, Analysis Controller, and File Relationships. These are tasked with handling requests on project uploads, project information management, project file and structure analysis, and file relationship management, respectively.
- View: The View component displays necessary information via interface packages such as Project Upload View, File Relationship View, Impact Analysis View, and Model View. These views facilitate the user interface for Simulink project uploads, file relationship management, impact analysis execution, and the visualization of changes and impacts within a specific project, respectively.

Fig. 15 and 16 display Simulysis screens, showcasing a standard Simulink file. The screen is divided into three sections, each marked with numbers for straightforward reference. The three sections are described as follows.

- Section 1: This section displays the differences between two specified file versions within the Simulink project. It features a search bar and tabs labeled "Added", "Changed", "Deleted", and "Impacted". These tabs categorize the systems, which are listed with an option to adjust their visibility.
- Section 2: This section outlines the functionalities associated with the WAVE-CIA method. It comprises two components: a dropdown list for selecting the *depth* of the Core set computation, and a button to initiate the WAVE-CIA analysis.
- Section 3: This section displays the Model View of the chosen Simulink file. It illustrates the Simulink systems (blocks) using the data of blocks and lines extracted from Simulink files, as detailed in Section III-B1. In this view, systems that have been added are highlighted in green; changed systems are marked in orange; and impacted systems are indicated in cyan. Other systems are either not filled or their color is determined by their properties.

V. Experiments

To assess the practical utility and efficacy of Simulysis, we have conducted tests using various open-source projects. These projects, sourced from GitHub[3], have a Git history of committed changes to Simulink source files. The projects include the following:

[2]https://github.com/KaoSon2004/Simulysis-CIA

[3]https://github.com/

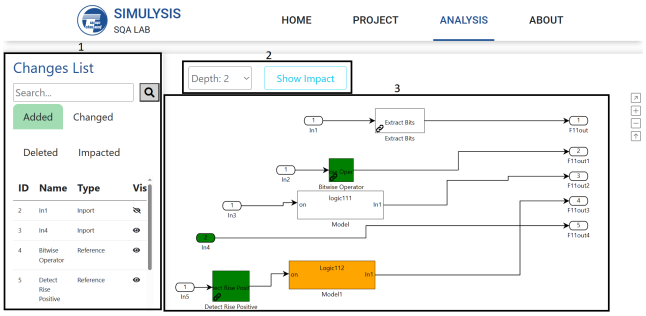


Fig. 15. An example of Simulink file in Simulysis.

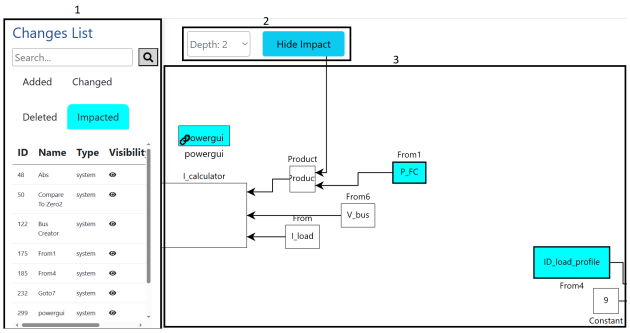


Fig. 16. An example of impacted systems in Simulysis.

- **Intelligent-EMS**[4]: This project aims to explore the feasibility of a Machine Learning-based Energy Management System (EMS) for an inboard DC-grid. The main energy source is hydrogen (Fuel-Cell), supplemented by a battery for quick load adjustments. In this project, Simulink is utilized for simulating a DC chip. For our experiment, we employed commit 5d200f (representing the old version) and commit 72216a (representing the new version).
- **Failure Detection and Fault Tolerant Control of a Jet-Powered Flying Humanoid Robot**[5] (called Failure Detection): This project involves Simulink source code used to conduct experiments for the related paper [42]. In our experiments, we utilized commit f0ed42 to represent the old version and commit 9427f4 for the new version.
- **FMT-Model**[6]: The Firmament Model (FMT Model) is a crucial part of the Firmament Autopilot project. This model provides a thorough simulation framework, along with algorithm libraries that have been carefully developed using MATLAB/Simulink. For our experiments, we referred to commit 37b5ae to represent the old version and commit f3c154 for the new version.
- **Reinforcement Learning for Control of Valves**[7] (called Reinforcement Learning): This project leverages Simulink to investigate the application of Reinforcement Learning for optimal control of non-linear systems. We conducted our experiments using two different versions of the system, represented by commit 171ef7 (the older version) and commit 25dc044 (the newer version).
- **Kugle**[8]: This project employs Simulink to make a self-balancing robot equipped with three omni wheels. Our experiments were conducted using two versions of the system, represented by commit 3be7e7 (the older version) and commit b1ee16 (the newer version).
- **Propulsion System Model**[9]: This project employs Simulink in the design of a Hybrid Motor Nitrous Oxide Rocket Propulsion System. Our experiments were conducted using two versions of the system, represented by commit 1994f02 (the older version) and commit 7aba367 (the newer version).

The five Simulink projects were chosen based on their increasing size and complexity, ranging from small and simple to large and complex, to comprehensively evaluate the effectiveness of the proposed method. These projects are well-known within the Simulink community, ensuring the results are relevant and applicable to commonly used projects. Additionally, the selected projects are complex enough to simulate real-world scenarios, demonstrating that the proposed method can be effectively applied in practical settings. By selecting projects with varying sizes and complexities, we ensure a thorough and representative assessment of the Simulysis method's performance in actual applications, implicitly suggesting its viability for real-world projects.

The experiments are performed on a server with the following configuration: Intel® Core™ i5-7300HQ @ 2.50 GHz, RAM: 16GB RAM. In our experiments, we set *depth = 3* for the core set computation. We checked the accuracy of the results by cross-referencing them with MATLAB's Simulink tool and calculated the following key metrics to evaluate the effectiveness of the Simulysis method: the number of changed systems (blocks), the number of impacted systems, the total number of systems in the new project version, the percentage of impacted systems in the new project version, the amount of memory required, and the time taken to perform the experiments. The results of these experiments are presented in Table VI. In this table, the columns are as follows:

- **Project**: the project under consideration.
- **CB**: number of changed blocks between the two versions.
- **IB**: number of impacted blocks in the new version.
- **TB**: total number of blocks in the new project version.
- **%**: percentage of impacted blocks within the new project (= IB/TB).
- **Mem (MB)**: amount of the required memory (in MB).
- **Time (s)**: The time (in seconds) taken to compute the impact set.

[4]https://github.com/dfpasdel/Intelligent_EMS
[5]https://github.com/ami-iit/paper_nava_2023_icra_fault-control-ironcub
[6]https://github.com/Firmament-Autopilot/FMT-Model/
[7]https://github.com/Rajesh-Siraskar/Reinforcement-Learning-for-Control-of-Valves/
[8]https://github.com/mindThomas/Kugle-MATLAB
[9]https://github.com/icl-rocketry/PropulsionSystemModel

TABLE VI. EXPERIMENTAL RESULTS

Project	CB	IB	TB	%	Mem (MB)	Time (s)
Intelligent-EMS	88	126	397	31.73	1.31	0.123
Failure Detection	16	26	1059	2.46	4.33	0.35
FMT-Model	2085	2668	9700	27.51	16.64	1.911
Reinforcement Learning	118	156	396	39.39	0.65	0.232
Kulge	263	322	1240	25.97	2.72	0.288
Propulsion	189	269	1115	24.13	5.78	0.767

Based on the experimental results presented in Table VI, we can make the following observations:

- For the "Intelligent-EMS" project, out of 397 blocks, 23 are impacted, which constitutes 5.79% of the total.
- For the "Failure Detection" project, out of 1059 blocks, 26 are impacted, which represents 2.46% of the total.
- For the "FMT-Model project", out of 9700 blocks, 289 are impacted, which represents 2.98% of the total.
- For the "Reinforcement Learning" project, out of 396 blocks, 50 are impacted, which represents 12.63% of the total.
- For the "Kugle", out of 1240 blocks, 76 are impacted, which represents 6.13% of the total.
- For the "Propulsion", out of 1115 blocks, 75 are impacted, which represents 6.73% of the total.
- In comparing the change calculation results, we found them to be accurate when cross-checked with MATLAB's Simulink model comparison tool. For the impact analysis, we tested our implementation and confirmed it aligns correctly with the WAVE-CIA algorithm. However, the correctness of the WAVE-CIA algorithm itself is beyond the scope of this discussion.
- For the largest project (FMT-Model), which comprises a total of 9700 systems, the memory and time consumption is 24.25 MB and 14.79 seconds, respectively. Given the current capabilities of memory and processing, we consider this to be acceptable. This demonstrates the practical applicability of Simulysis for real-world projects.
- While it appears that the memory and time usage may be proportional to the number of changed and impacted systems, there is no definitive evidence to support this observation.

VI. RELATED WORKS

Numerous studies have been conducted on the Change Impact Analysis (CIA) issue, specifically about Simulink projects and more generally, software projects. In this section, we focus solely on the most relevant studies and tools [37], [5], [6], [40], [43], [44], [3], [45].

In the realm of model-based testing, many researchers have made significant contributions to UML impact analysis. In 2003, Briand et al. introduced a UML model-based approach to impact analysis that can be implemented prior to any changes [43]. Initially, they ensure the consistency of the UML diagrams. Subsequently, they identify the modifications between two distinct versions of a UML model. Following this, they determine the model elements that are directly or indirectly affected by these changes using impact analysis rules (formulated in Object Constraint Language). Our interest aligns with Briand's in the area of software change impact analysis. However, our approach involves analyzing the source code of Simulink projects to construct the call graph and applying the WAVE-CIA method to compute the impact set.

In 2010, Fourneret et al. introduced an automated model-based impact analysis [44]. Fourneret's approach involves a set of algorithms that identify dependent and impacted elements following the detection of changes in a UML/OCL Statechart diagram. This allows an engineer to pinpoint the critical aspects of the software and determine which functionalities require regression testing. While our interests align with Fourneret's in the area of software system regression testing, our focus is specifically on the change impact analysis of Simulink projects.

In 2010, Li et al. introduced the WAVE-CIA method, a novel approach for calculating the impact set derived from the call graph of various types of software projects [3]. Our work aligns with this interest in change impact analysis. However, our application of the WAVE-CIA method is specifically tailored to Simulink projects. We construct the call graph using Simulink files and subsequently apply the WAVE-CIA method to these generated call graphs.

Diffplug [37] conducts impact analysis of Simulink projects using a straightforward method known as signal tracing. This method allows users to select a block and trace its signal, enabling them to determine the potential impact of a block change. However, Diffplug necessitates that users manually navigate through each level of the model to identify the impacted target inputs and/or outputs. Conversely, Simulysis automates this process by analyzing the project, calculating changes, and computing the impact set using the WAVE-CIA method.

In 2017, Rapos and Cordy introduced SimPact, a method and tool for analyzing the impact of changes in Simulink models [5]. SimPact's impact analysis of a Simulink model involves two stages: change isolation and the determination of the impact of changes on test values. Initially, SimPact identifies the differences between two versions of a model. Subsequently, it assesses the potential impact of these changes

on inputs and outputs by traversing the entire model hierarchy, tracing, and highlighting all potential impacts from a specific block at all levels. While our interest in the impact analysis of Simulink projects aligns with that of SimPact, there are key differences between SimPact and Simulysis. In the change isolation phase, SimPact utilizes the model differencing script of Rapos and Cordy [46], which employs the internal Simulink comparison operator. Conversely, Simulysis directly analyzes the Simulink project to construct the call graph and compares the two call graph versions to identify the change set. In the impact analysis phase, SimPact depends on the caller-callee relationship to identify the impact set, which is fundamentally different from Simulysis, which uses the WAVE-CIA method to determine the impact set.

In 2020, Bennett et al. introduced the Boundary Diagram Tool (BDT), a method designed for change impact analysis of large Simulink projects within embedded systems [6]. BDT conducts impact analysis by tracing changes throughout the Simulink model. Our work aligns with this interest in Simulink project impact analysis. However, our approach differs as we directly analyze Simulink projects to construct the call graph and utilize the WAVE-CIA method to compute the impact set.

In 2021, Jaskolka et al. [40] conducted an examination of real industrial software repositories and their version control systems to gain insights into potential changes in Simulink. Their intention was to guide the usage of Simulink in industrial practices and understand the distribution of change types in Simulink projects. However, Simulysis differs fundamentally in its objectives, as it primarily aims to compute the impact set.

In 2023, Tran et al. developed CIA4CS, a method specifically designed for Change Impact Analysis (CIA) of C# projects, based on the WAVE-CIA approach [45]. CIA4CS fills a significant void in existing methodologies, considering the widespread use of the C# language in the software industry. The method utilizes static code analysis to build dependency graphs, which then facilitates the application of the WAVE-CIA. The authors have integrated CIA4CS into a tool and conducted tests on a variety of C# projects. Their experiments demonstrated the tool's efficiency in analyzing project components, dependencies, and the elements impacted across different versions. While our interests align with Tran's in terms of the CIA and the WAVE-CIA method, our focus is specifically on the CIA of Simulink projects.

VII. Conclusion

We have introduced Simulysis, a method designed for the change impact analysis of Simulink projects. When provided with two versions of a Simulink project, Simulysis performs the CIA by constructing the call graph from the files, identifying the changes, and calculating the impact set using the WAVE-CIA method. We have incorporated this method into a tool, also named Simulysis, and conducted several experiments on open-source projects, yielding promising results. Further discussion about the method is provided in the paper.

Looking ahead, we envision several enhancements to Simulysis to augment its capabilities. Firstly, we are actively working on improving the speed of call graph construction and refining the algorithm used for impact set determination. Secondly, we plan to upgrade the tool's user interface to facilitate better visualization of large-scale Simulink projects. Lastly, we aim to develop additional features such as report exporting and system integration support to better accommodate user environments. Through these advancements, our goal is to offer users an efficient, user-friendly tool that bolsters the role of Simulysis in ensuring effective software quality assurance for Simulink projects.

Acknowledgments

This research was funded by the research project QG.25.09 of Vietnam National University, Hanoi.

References

[1] B. Li, X. Sun, H. Leung, and S. Zhang, "A survey of code-based change impact analysis techniques," *Software Testing, Verification and Reliability*, vol. 23, no. 8, pp. 613–646, 2013. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/stvr.1475

[2] M. Kretsou, E.-M. Arvanitou, A. Ampatzoglou, I. Deligiannis, and V. C. Gerogiannis, "Change impact analysis: A systematic mapping study," *Journal of Systems and Software*, vol. 174, p. 110892, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S016412122030282X

[3] B. Li, Q. Zhang, X. Sun, and H. Leung, "Wave-cia: A novel cia approach based on call graph mining," in *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, ser. SAC '13. New York, NY, USA: Association for Computing Machinery, 2013, p. 1000–1005. [Online]. Available: https://doi.org/10.1145/2480362.2480554

[4] K. A. Alam, R. Ahmad, A. Akhunzada, M. H. N. M. Nasir, and S. U. Khan, "Impact analysis and change propagation in service-oriented enterprises: A systematic review," *Information Systems*, vol. 54, pp. 43–73, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306437915001179

[5] E. J. Rapos and J. R. Cordy, "Simpact: Impact analysis for simulink models," in *2017 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, 2017, pp. 489–493.

[6] B. Mackenzie, V. Pantelic, G. Marks, S. Wynn-Williams, G. Selim, M. Lawford, A. Wassyng, M. Diab, and F. Weslati, "Change impact analysis in simulink designs of embedded systems," ser. ESEC/FSE 2020. New York, NY, USA: Association for Computing Machinery, 2020, p. 1274–1284. [Online]. Available: https://doi.org/10.1145/3368089.3417060

[7] Bohner, "Impact analysis in the software change process: a year 2000 perspective," in *1996 Proceedings of International Conference on Software Maintenance*, 1996, pp. 42–51.

[8] O. Bouchaala, M. Yangui, S. Tata, and M. Jmaiel, "Dat: Dependency analysis tool for service based business processes," in *2014 IEEE 28th International Conference on Advanced Information Networking and Applications*, 2014, pp. 621–628.

[9] S. Lehnert, "A taxonomy for software change impact analysis," in *Proceedings of the 12th International Workshop on Principles of Software Evolution and the 7th Annual ERCIM Workshop on Software Evolution*, ser. IWPSE-EVOL '11. New York, NY, USA: Association for Computing Machinery, 2011, p. 41–50. [Online]. Available: https://doi.org/10.1145/2024445.2024454

[10] S. Basu, F. Casati, and F. Daniel, "Toward web service dependency discovery for soa management," in *2008 IEEE International Conference on Services Computing*, vol. 2, 2008, pp. 422–429.

[11] D. Romano and M. Pinzger, "Using vector clocks to monitor dependencies among services at runtime," in *Proceedings of the International Workshop on Quality Assurance for Service-Based Applications*, ser. QASBA '11. New York, NY, USA: Association for Computing Machinery, 2011, p. 1–4. [Online]. Available: https://doi.org/10.1145/2031746.2031748

[12] S. Qi, B. Li, C. Liu, X. Wu, and R. Song, "A trust impact analysis model for composite service evolution," in *Proceedings of the 2012 19th Asia-Pacific Software Engineering Conference - Volume 01*, ser. APSEC '12. USA: IEEE Computer Society, 2012, p. 73–78. [Online]. Available: https://doi.org/10.1109/APSEC.2012.30

[13] D. Zhao, S. Liu, L. Wu, R. Wang, and X. Meng, "Hypergraph-based service dependency resolving and its applications," in *2012 IEEE Ninth International Conference on Services Computing*, 2012, pp. 106–113.

[14] Y. Wang, J. Yang, W. Zhao, and J. Su, "Change impact analysis in service-based business processes," *Service Oriented Computing and Applications*, vol. 6, no. 2, p. 131–149, 2012.

[15] Q. Yu, X. Liu, A. Bouguettaya, and B. Medjahed, "Deploying and managing web services: issues, solutions, and directions," *The VLDB Journal*, vol. 17, no. 3, p. 537–572, 05 2008. [Online]. Available: https://doi.org/10.1007/s00778-006-0020-3

[16] P. Kumar and Ratneshwer, "A review on dependency analysis of soa based system," 2014. [Online]. Available: https://api.semanticscholar.org/CorpusID:11112313

[17] S. Black, "Computing ripple effect for software maintenance," *Journal of Software Maintenance and Evolution: Research and Practice*, vol. 13, no. 4, pp. 263–279, 2001. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/smr.233

[18] H. Xiao, J. Guo, and Y. Zou, "Supporting change impact analysis for service oriented business applications," in *International Workshop on Systems Development in SOA Environments (SDSOA'07: ICSE Workshops 2007)*, 2007, pp. 6–6.

[19] S. Wang and M. A. Capretz, "A dependency impact analysis model for web services evolution," in *2009 IEEE International Conference on Web Services*, 2009, pp. 359–365.

[20] ——, "Dependency and entropy based impact analysis for service-oriented system evolution," in *2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*, vol. 1, 2011, pp. 412–417.

[21] B. Li, X. Sun, H. Leung, and S. Zhang, "A survey of code-based change impact analysis techniques," *Software Testing, Verification and Reliability*, vol. 23, no. 8, pp. 613–646, 2013. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/stvr.1475

[22] L. Westfall, "Bidirectional requirements traceability," *Software Quality Professional Magazine*, vol. 10, 2007. [Online]. Available: https://api.semanticscholar.org/CorpusID:107717465

[23] C. Boldyreff, E. Burd, R. Hather, M. Munro, and E. Younger, "Greater understanding through maintainer driven traceability," in *WPC '96. 4th Workshop on Program Comprehension*, 1996, pp. 100–106.

[24] S. Pfleeger and S. Bohner, "A framework for software maintenance metrics," in *Proceedings. Conference on Software Maintenance 1990*, 1990, pp. 320–327.

[25] H. K. Dam and A. Ghose, "Mining version histories for change impact analysis in business process model repositories," *Computers in Industry*, vol. 67, pp. 72–85, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0166361514001833

[26] T. Zimmermann, P. Weibgerber, S. Diehl, and A. Zeller, "Mining version histories to guide software changes," in *Proceedings. 26th International Conference on Software Engineering*, 2004, pp. 563–572.

[27] H. Kagdi, M. L. Collard, and J. I. Maletic, "A survey and taxonomy of approaches for mining software repositories in the context of software evolution," *J. Softw. Maint. Evol.*, vol. 19, no. 2, p. 77–131, mar 2007. [Online]. Available: https://doi.org/10.1002/smr.344

[28] H. Khanh Dam, "Predicting change impact in web service ecosystems," *International Journal of Web Information Systems*, vol. 10, no. 3, pp. 275–290, mar 2014. [Online]. Available: https://doi.org/10.1108/IJWIS-03-2014-0006

[29] W. Fdhila, S. Rinderle-Ma, and C. Indiono, "Memetic algorithms for mining change logs in process choreographies," in *Service-Oriented Computing*, X. Franch, A. K. Ghose, G. A. Lewis, and S. Bhiri, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 47–62.

[30] Broadcom, "Ca harvest software change manager," accessed date: 2024-02-15. [Online]. Available: https://www.broadcom.com/products/software/service-management/harvest-software-change-manager

[31] J. Buckner, J. Buchta, M. Petrenko, and V. Rajlich, "Jripples: A tool for program comprehension during incremental change," in *Proceedings of the 13th International Workshop on Program Comprehension*, ser. IWPC '05. USA: IEEE Computer Society, 2005, p. 149–152. [Online]. Available: https://doi.org/10.1109/WPC.2005.22

[32] CoderGears, "Jarchitect :: Java static analysis and code quality tool," accessed date: 2024-02-15. [Online]. Available: https://www.jarchitect.com

[33] SonarSource, "Clean code: Writing clear, readable, understandable; reliable quality code — sonar," accessed date: 2024-02-15. [Online]. Available: https://www.sonarsource.com/products/sonarqube/

[34] Checkmarx, "Application security testing company — software security testing solutions — checkmarx," accessed date: 2024-02-15. [Online]. Available: https://checkmarx.com/

[35] R. Hat, "Ansible is simple it automation," accessed date: 2024-02-15. [Online]. Available: https://www.ansible.com

[36] Atlassian, "Jira," accessed date: 2024-02-15. [Online]. Available: https://www.atlassian.com/software/jira

[37] Diffplug, "Free simulink viewer and differ — diffplug," accessed date: 2024-02-15. [Online]. Available: https://www.diffplug.com/features/simulink

[38] MathWorks, *Simulink - Simulation and Model-Based Design - MATLAB*, accessed date: 2024-02-25. [Online]. Available: https://www.mathworks.com/products/simulink.html

[39] ——, *Matlab*, accessed date: 2024-02-25. [Online]. Available: https://www.mathworks.com/products/matlab.html

[40] M. Jaskolka, V. Pantelic, A. Wassyng, M. Lawford, and R. Paige, "Repository mining for changes in simulink models," in *2021 ACM/IEEE 24th International Conference on Model Driven Engineering Languages and Systems (MODELS)*, 2021, pp. 46–57.

[41] R. Tarjan, "Depth-first search and linear graph algorithms," in *12th Annual Symposium on Switching and Automata Theory (swat 1971)*, 1971, pp. 114–121.

[42] G. Nava and D. Pucci, "Failure detection and fault tolerant control of a jet-powered flying humanoid robot," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 12 737–12 743.

[43] L. Briand, Y. Labiche, and L. O'Sullivan, "Impact analysis and change management of uml models," in *International Conference on Software Maintenance, 2003. ICSM 2003. Proceedings.*, 2003, pp. 256–265.

[44] E. Fourneret and F. Bouquet, "Impact analysis for uml/ocl statechart diagrams based on dependence algorithms for evolving critical software," sep 2010, accessed date: 2024-02-15. [Online]. Available: https://publiweb.femto-st.fr/tntnet/entries/120/documents/author/data

[45] H. V. Tran, N. T. M. Loan, D. D. Kien, N. H. Trang, L. V. Huy, and P. N. Hung, "Cia4cs: A method for change impact analysis of c# projects," *Journal on Information Technologies and Communications*, vol. 2024, no. 01, 11 2023. [Online]. Available: https://doi.org/10.1145/2480362.2480554

[46] E. J. Rapos and J. R. Cordy, "Examining the co-evolution relationship between simulink models and their test cases," in *Proceedings of the 8th International Workshop on Modeling in Software Engineering*, ser. MiSE '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 34–40. [Online]. Available: https://doi.org/10.1145/2896982.2896983

Feature Selection and Classification of Microarray Datasets Based on an Improved Binary Harris Hawks Optimization Algorithm

Guoxia LI[1], Wen SHI[2], Jingyu ZHANG[3], Zhixia GU[4], Jixiang XU[5], Yueyue LI[6], Yaxing SUN[7]
School of Information Engineering, Tianjin University of Commerce, Tianjin 300134, China[1,2,4,5,6,7]
Tianjin NanKai Hospital, Tianjin 300100, China[3]

***Abstract*—High-dimensional microarray datasets are prone to the "curse of dimensionality" due to feature redundancy, which impairs the performance of machine learning models, and feature selection is the key to addressing this issue. This study proposes an Improved Binary Harris Hawks Optimization algorithm (IBHHO) for feature selection in high-dimensional microarray data. Core innovations comprise: i) a hybrid filter-wrapper framework integrating a filter method (ReliefF), a wrapper method (HHO) and a classifier (SVM) to simultaneously optimize ReliefF parameters, SVM hyperparameters, and feature subsets; ii) a differentiated exploration–exploitation strategy leveraging HHO's two-stage behavior (global parameter optimization during exploration; feature refinement and local parameter tuning during exploitation); and iii) an elite feature guidance strategy that reduces redundant exploration and accelerates convergence via fixed key-feature anchor points. Experiments conducted on eight public microarray datasets demonstrate that IBHHO reduces feature counts while improving classification accuracy, achieving comprehensive performance superior to benchmark algorithms. Consequently, IBHHO offers an efficient feature selection framework for high-dimensional biomedical data analysis.**

***Keywords*—Microarray dataset; feature selection; parameter optimization; ReliefF**

I. Introduction

Technological advancements in the contemporary era have driven revolutionary transformations in data acquisition methodologies, while also engendering the proliferation of redundant or misleading data—this may inadvertently compromise the performance of learning models. Furthermore, coupled with the "curse of dimensionality" [1], the substantial volume of features in datasets frequently induces overfitting in machine learning frameworks.

A domain of particular interest is the analysis of high-dimensional microarray datasets for gene expression profiling, which facilitates the identification of cancer-related genes. Microarray data (obtained from microarray experiments) is typically presented as a two-dimensional table, with rows representing gene expression levels and columns corresponding to samples [2]. Notably, these datasets are characterized by a small number of samples juxtaposed against hundreds of thousands of genes. While microarray datasets have been widely used for cancer classification [3], a large number of genes are irrelevant to disease classification—posing a challenge for understanding target diseases and underscoring the crucial role of feature selection in reducing data dimensionality [4].

*Corresponding authors.

The objective of feature selection for microarray data is to identify a set of genes that best discriminates between biological sample types, eliminating irrelevant and redundant information to enhance the efficiency and accuracy of cancer classification algorithms. According to feature evaluation patterns, feature selection algorithms are broadly categorized into three types: filter methods, wrapper methods, and hybrid methods [5].

Filter methods employ fast standard metrics to evaluate feature importance without classifier feedback, offering low computational complexity but being agnostic to classifier types. In contrast, wrapper methods leverage classifier feedback for feature subset selection—their selection process and performance are biased toward the specific classifier, and while they achieve superior classification accuracy, they incur higher computational overhead. The hybrid method integrates wrapper and filter approaches: in the initial stage, a filter method quickly screens out irrelevant features to reduce dimensionality; subsequently, a wrapper method explores the reduced feature space for a more concise subset [6]. It balances computational efficiency and classification accuracy—slightly slower than pure filters but with better classification performance, and significantly less complex than standalone wrappers while achieving comparable or superior results.

Recently, hybrid feature selection strategies have gained traction for high-dimensional datasets (e.g., [7], [8]), typically using a filter to preselect top n features before a wrapper searches for the optimal subset. However, this traditional two-stage strategy has obvious limitations: the filtering stage relies solely on statistical correlation between features and the target, ignoring the classifier's learning needs—potentially discarding features crucial for classification but with insignificant statistical indicators (which cannot be reintroduced in the wrapping stage). Additionally, the manually preset n creates a trade-off: too small restricts the wrapper's search range, while too large increases computational load—this defect is particularly prominent in microarray data. Therefore, this study proposes integrating the filter's initial feature screening into the wrapper stage via a dynamic optimization mechanism. Filter feature evaluation receives classifier feedback to avoid misjudgment; unselected features are retained in a candidate pool for reevaluation during wrapper iteration, enabling dynamic recall of useful features. Meanwhile, filter subset size parameters adapt to different datasets.

Support Vector Machine (SVM) [9] is a widely used

classification technique with proven effectiveness [10]. Beyond feature selection, SVM parameter configuration significantly impacts accuracy, and the core challenge in enhancing SVM's performance lies in simultaneous optimization of feature subsets (discrete 0/1 variables) and SVM parameters (continuous values)—a mixed-variable optimization problem [11].

Metaheuristic algorithms (e.g., Harris Hawks Optimization, HHO [12]) are well-suited for wrapper-based feature selection due to computational efficiency and global search capabilities. HHO mimics Harris hawks' cooperative hunting (exploration, transition, exploitation): the exploration phase avoids local optima via global solution space search, while the exploitation phase refines solution quality through precise local adjustments. Accordingly, this study integrates filter parameter optimization into HHO's exploration phase (aligning feature rankings with classification objectives) and feature subset refinement into the exploitation phase (eliminating redundancy, retaining critical features, and adapting classifier parameters). This phased integration realizes dynamic coordination between parameter tuning and feature screening. Traditional HHO-based feature selection (binary encoding of candidate subsets) lacks clear directional anchors, leading to excessive ineffective searches in redundant feature regions—slowing convergence and risking missed optimal solutions. To address this, we propose an "elite feature guidance mechanism": key features in the top-performing subset of each iteration are marked as "elite" (fixed as selected in subsequent optimizations). These features act as search anchors, reducing ineffective exploration, accelerating convergence to the global optimal subset, and improving selection efficiency and accuracy. Given HHO's advantages in avoiding local optima, this paper proposes an Improved Binary HHO (IBHHO) for feature selection in high-dimensional microarray datasets. The main contributions are summarized as follows:

1) Hybrid feature selection framework for joint optimization of high-dimensional microarray data: A filter-wrapper integrated framework enabling concurrent optimization of ReliefF parameters, SVM hyperparameters, and feature subsets. ReliefF's subset size parameter is adaptively tuned based on dataset feature dimensionality.

2) Co-optimization strategy based on HHO's two-stage characteristics: A differentiated optimization strategy leveraging HHO's phases: the exploration phase optimizes ReliefF and SVM parameters for comprehensive feature space exploration; the exploitation phase refines feature subsets and fine-tunes classifier parameters—significantly enhancing overall classification performance.

3) Elite feature guidance strategy: An "Elite Feature Guidance Strategy" that fixes key features (labeled 1) as strict binary 1s. These anchored features steer the algorithm to converge toward verified optimal subsets, mitigating redundant exploration, accelerating convergence, and improving selection accuracy/efficiency.

IBHHO addresses incoordination between filter and wrapper stages in traditional hybrids and optimization drawbacks of conventional HHO. Tested on eight high-dimensional microarray datasets, it outperforms existing popular meta-heuristic methods—verifying its effectiveness for microarray data feature selection.

The study is structured as follows: Section II reviews related research on feature selection for microarray datasets; Section III elaborates on basic method principles; Section IV introduces the optimization problem model; Section V describes IBHHO's application to feature selection; Section VI evaluates IBHHO using eight public datasets; and Section VII concludes the study.

II. Related Work

A. Feature Selection Methods

Filter-based feature selection methods assess feature importance exclusively based on data-intrinsic properties, disregarding the influence of learning algorithms or classifiers on feature utility. Prominent examples include, in reference [13], employed the ReliefF algorithm to select the top 3% feature subsets in the initial screening stage of feature selection for high-dimensional microarrays. In [14], the authors utilized the Simulated Kalman Filter algorithm to select 1% of features in microarray feature selection. In [15], the authors adopted the SLI-γ filter method to choose the top 1% most relevant features.

Wrapper-based feature selection methods incorporate learning algorithms as integral components, directly leveraging classifier performance as the evaluation metric for feature subset optimization. Wrapper methods typically have higher computational costs than filter methods because they rely on iterative evaluation of classifiers. Prominent examples include, in reference [16], proposes a wrapper feature selection method based on the Chimp Optimization Algorithm. In [17], the authors introduces a Multi-objective Binary Harris Hawks Optimization algorithm to directly optimize feature subsets as a wrapper approach. In [18], the authors employ the Genetic Algorithm as a wrapper method, using the accuracy of the KNN classifier as the fitness function.

For high-dimensional microarray datasets—where feature spaces often comprise thousands to tens of thousands of genes—filter-wrapper hybrid approaches offer distinct advantages over pure wrapper methods [19]. These hybrids integrate the dimensionality reduction efficiency of filter methods with the discriminative power of wrapper optimization, striking a balance between computational feasibility and classification performance.

Recently, feature selection methods based on hybrid filter wrappers have been widely used to speed up the selection process and improve classification performance. In reference [20], an altruism mechanism was integrated into the Whale Optimization Algorithm (WOA) to develop the AltWOA algorithm. First, Pasi Luukka's fuzzy entropy filtering method [21] was employed to screen the top 300 genes by entropy values, eliminating features with low class correlation. Subsequently, the improved WOA—incorporating altruistic behavior—used SVM classification accuracy as the fitness function to perform iterative optimization of the feature subset from the preselected 300 genes. Evaluation on eight high-dimensional microarray datasets demonstrates that this algorithm outperforms other algorithms in terms of accuracy. The hybrid feature selection framework proposed in [22] employs a two-stage strategy. In the filter stage, the minimal redundancy maximal relevance

algorithm integrates multiple filtering metrics—including ReliefF, chi-square test, and Kullback-Leibler divergence—to quantify gene relevance, selecting the top 50 features with the highest scores. Subsequently, in the wrapper stage, an improved gray wolf optimizer is deployed to search for the optimal feature combination within the prefiltered candidate set, using SVM classification accuracy as the fitness function. In [23], the authors propose a two-stage gene selection framework for microarray data, integrating anomaly detection with genetic algorithms. In the first stage, an autoencoder is employed to reduce the dimensionality of gene expression data, followed by one-class support vector machine (One-class SVM) for anomalous gene detection, where approximately 1% of features are selected as the candidate subset. The second stage then applies a guided genetic algorithm to the candidate genes, refining them into the final set of effective features through fitness evaluation based on classification accuracy. In [24], the authors propose a novel hybrid algorithm, TRF-WGHC, by integrating a ranking-based filter feature selection algorithm with a greedy hill-climbing algorithm for DNA microarray applications. In the first stage, specific ranking metrics—including information gain, gain ratio, and ReliefF—are employed to select the top n percent of genes, discarding those with scores below a predefined threshold. The second stage then utilizes an augmented greedy hill-climbing algorithm to search for the optimal feature subset from the remaining genes. The experimental part was comprehensively tested on 18 microarray datasets, demonstrating that the algorithm is simple and effective.

B. Joint Optimization

On one hand, the performance of Support Vector Machine (SVM) models is significantly influenced by parameter selection, and applying SVM presents two key challenges: selecting the optimal input feature subset and determining the best parameter values, which are interdependent—the chosen feature subset affects the optimal parameter values, and vice versa. On the other hand, the parameters of the filter method used also impact feature subset selection and classification performance. Therefore, this study focuses on synchronously optimizing three variables: SVM parameters, filter method parameters, and feature selection results. Within the synchronous optimization framework, the feature selection vector consists of binary elements—"1" indicates selecting the corresponding feature, while "0" excludes it [25]; the feature mask for selection is a discrete integer variable, whereas SVM classifier parameters and filter method parameters are continuous variables, making the synchronous optimization of feature selection and SVM parameters a mixed-variable optimization problem. In the study of reference [26], researchers discretized continuous parameters into integers and adopted the Firefly Algorithm to synchronously optimize feature selection and SVM parameters to improve the performance of traditional Chinese medicine prescription classification. In [27], the authors used the Genetic Algorithm with a binary encoding system to design chromosomes containing penalty parameter C, kernel function parameters, and feature masks, realizing the joint optimization of feature selection and parameters. In [25], the authors introduced the Salp Swarm Algorithm to address the joint optimization of feature selection and classifier parameter tuning, partitioning the solution space into a feature mask (discrete part) and SVM parameters (continuous part), which are optimized by binary SSA and standard SSA respectively. In this study, the IBHHO algorithm synchronously optimizes the three variables: during the exploration phase of the algorithm, the filter algorithm and SVM parameters are optimized using the original continuous HHO algorithm; during the exploitation phase, SVM parameters and feature subsets are optimized using both the continuous HHO algorithm and binary HHO algorithm.

C. Summary

In summary, existing filter-wrapper hybrid methods for microarray feature selection typically adopt a two-stage process: pre-screening a fixed number of features using filter methods, followed by wrapper optimization. This rigid architecture has two inherent flaws: 1) Static filtering thresholds cannot adapt to the dynamic changes in feature correlations within microarray data. When processing different microarray datasets, due to the lack of flexibility and adaptability, it is difficult to achieve effective optimization of feature selection and the performance of classification models. 2) Optimizing the filtering phase and the wrapper phase separately may result in a lack of close coordination between them. This can cause the feature subsets initially screened by the filtering algorithm to be affected by redundant features, or lead to the omission of key features due to the limitations of statistical evaluation, ultimately impairing the overall performance of the model. In contrast, this study proposes IBHHO algorithm. By integrating the filtering phase into the two-stage mechanism of HHO, it realizes the combination of the filtering phase and the wrapper phase in the iterative update of the HHO algorithm, while achieving the dynamic adaptive adjustment of the weight parameters of the filtering method. This enables the filtering phase to adaptively adjust the number of feature subsets according to the specific distribution of the dataset, thereby effectively avoiding the risks of over-filtering (losing discriminative features) and redundant retention (retaining irrelevant attributes).

III. Preliminary Knowledge

This section elaborates on the preliminary knowledge employed in this study. First, a feature ranking algorithm based on the ReliefF filter is introduced, which is used for the initial screening of feature subsets. Subsequently, the SVM classifier used in this study and the application process of the original HHO algorithm are presented.

A. ReliefF

A notable strength of ReliefF [28] in feature evaluation is its capacity to capitalize on local feature dependencies while still accounting for the global data distribution. This avoids sacrificing the global perspective by overemphasizing local details [29]. Its core principle involves assessing features according to their capability to discriminate between closely neighboring observation samples via feature values. Specifically, the evaluation unfolds iteratively: in each iteration, an observation sample is randomly drawn from the sample space; next, the k nearest neighbors belonging to the same category and the corresponding nearest neighbors from other categories are identified; subsequently, each feature's score is updated based on feature-value differences between the target

sample and its same-category and different - category nearest neighbors. The weights for each feature in distinguishing the target sample from its neighbors are computed using Eq. (1). Generally, a higher weight indicates greater importance for tasks like category classification, translating to higher utility in model construction and related processes.

$$W_f = W_f - \frac{1}{mk}\sum_{j=1}^{k}\mathrm{diff}_f(R_i, H_j) + \frac{1}{mk}\sum_{c\neq \mathrm{class}(R_i)}\frac{P(c)}{1-P(\mathrm{class}(R_i))}\sum_{j=1}^{k}\mathrm{diff}_f(R_i, M_j) \tag{1}$$

where, W_f is the ReliefF weight of feature f, k denotes the number of nearest neighbor samples, and m represents the number of iterations. R_i is the random sample in iteration i. H_j and M_j are the nearest neighbor samples from the same class and different classes of R_i, respectively. $P(c)$ is the prior probability of class c, which is typically determined based on the training samples fed into the ReliefF algorithm. $\mathrm{diff}_f(\cdot,\cdot)$ is the value difference of feature f between two observations.

For the ReliefF filtering method employed in the IBHHO algorithm proposed in this study, after ranking features by weights using k-nearest neighbors, the top SN features (where, SN denotes the number of features selected by ReliefF) are chosen as the feature subset output in the initial screening stage. The setting equations for k and SN are presented in Eq. (2) and Eq. (3).

$$k = \lfloor \vec{p}\cdot k_{\max} + 0.5 \rfloor \tag{2}$$

$$SN = \lfloor \vec{p}\cdot D + 0.5 \rfloor \tag{3}$$

where, $\vec{p}$ represents the position vector of k, $k_{\max}$ is typically set to 10, $\vec{p}\cdot k_{\max}$ means scaling each element of vector $\vec{p}$ by $k_{\max}$ times, and $\lfloor \vec{p}\cdot k_{\max} + 0.5 \rfloor$ means rounding $\vec{p}\cdot k_{\max}$ to the nearest integer. D represents the feature dimension of the dataset, and $\vec{p}\cdot D$ means scaling each element of vector is scaled by D times to realize parameter adjustment based on the data dimension.

During the filtering stage, the ReliefF algorithm is employed to assess the significance of each feature. Features are then ranked in descending order of their computed importance scores, and the top SN features are forwarded to the wrapper stage. Setting an excessively low threshold may lead to the exclusion of critical features, whereas an overly high threshold could retain numerous redundant or irrelevant features. To address this challenge, our approach dynamically adjusts the threshold, enabling the selection of feature subsets with varying sizes according to the intrinsic importance distribution of features across diverse datasets.

B. SVM

The core idea of Support Vector Machines (SVM) is to map training data into a higher-dimensional space and separate the categories of training data by establishing an optimal hyperplane, thereby transforming non-linearly separable data in the input feature space into linearly separable data in the high-dimensional feature space.

When using SVM as a classifier, the parameter C serves as a key factor in controlling the model's complexity and the severity of misclassification penalties, thereby balancing the model's bias and variance. Additionally, the kernel function parameter determines the nonlinear transformation from the input space to the high-dimensional space, where the hyperplane used for class separation is identified. This study selects the RBF kernel function based on its simplicity and efficiency, as shown in Eq. (4):

$$\varphi(x_i - x_j) = \exp\left(-\gamma\|x_i - x_j\|^2\right), \gamma > 0 \tag{4}$$

where, γ represents the parameter of kernel function. The parameters of support vector machine (SVM) to be optimized include penalty parameter C and kernel function parameter γ.

C. Harris Hawk Optimization

The HHO algorithm consists of three distinct phases: the exploration phase, the transition phase from exploration to exploitation, and the exploitation phase.

Proposed in [12], the Harris Hawk Optimization algorithm (HHO) is a bio-inspired swarm intelligence optimization algorithm renowned for its robust global search capability and high optimization precision. In this framework, the prey (rabbits in natural hunting scenarios) symbolizes the optimal solution with the highest fitness value in the current iteration. The algorithm operates through two overarching stages: exploration and exploitation, with the transition phase mediating between them. During the exploration phase, the algorithm initializes control parameters (escape energy E) to survey the solution space and locate potential prey regions. The exploitation phase comprises four distinct attack strategies, dynamically adjusted based on the prey's perceived "energy" (solution quality) and escape probability. These strategies simulate hierarchical hunting behaviors, enabling progressive refinement of the optimal solution.

1) Exploration phase: When hunting prey, Harris hawks adopt two distinct exploration strategies. Candidate solutions are designed to approach the prey as closely as possible, with the optimal solution representing the target prey itself. First, Harris hawks select a landing point by considering the positions of other hawks and their prey. In the second method, hawks wait on random tall trees. These two approaches are simulated with equal probability q using Eq. (5):

$$x(t+1) = \begin{cases} x_{\mathrm{r}}(t) - r_1\left|x_{\mathrm{r}}(t) - 2r_2x(t)\right| & q \geq 0.5 \\ x_{\mathrm{p}}(t) - x_{\mathrm{m}}(t) - r_3\left(L + r_4(U-L)\right) & q < 0.5 \end{cases} \tag{5}$$

where, the vector $\mathbf{x}(t)$ is the current position of the hawk, $\mathbf{x}(t+1)$ is the hawk's position in the next iteration, $\mathbf{x}_{\mathrm{r}}(t)$ is a randomly selected position from the Harris hawk population, $\mathbf{x}_{\mathrm{p}}(t)$ is the prey position in the current iteration, r_1, r_2, r_3, r_4 are random numbers with a $(0,1)$ distribution, q is the transition factor controlling the two strategies, and L/U are the lower/upper bounds of variables. $\mathbf{x}_{\mathrm{m}}(t)$ is the average

position of the current hawk population, is calculated as Eq. (6):

$$x_{\mathrm{m}}(t) = \frac{1}{N}\sum_{i=1}^{N} x_i(t) \tag{6}$$

where $\mathbf{x}_i(t)$ is the position of each hawk in iteration t, and N denotes the total number of hawks.

2) Transition from exploration to exploitation: The algorithm switches from exploration to exploitation based on the prey's running or escape energy, as defined by Eq. (7):

$$E = 2E_0\left(1 - \frac{t}{Max_iter}\right) \tag{7}$$

where, E represents the prey's escape energy, E_0 is the initial energy state, which randomly varies within $(-1, 1)$ at each iteration. When $|E| \geq 1$, global search is performed; otherwise, the exploitation phase begins.

3) Exploitation phase: During the exploitation phase $|E| < 1$, Harris hawks raid and capture prey while the prey avoids predation. HHO determines the appropriate position update strategy from the following four attack modes based on the random number $r \in (0, 1)$, parameter E, and the strategy determinant $|E|$: when $|E| \geq 0.5$, Harris hawks choose the soft besiege strategy; otherwise, they adopt the hard besiege strategy.

a) Soft besiege: When $r \geq 0.5$, Harris hawks can capture the prey; otherwise, the hunt fails. When $r \geq 0.5$ and $|E| \geq 0.5$, the prey has sufficient energy to jump to avoid predation, and Harris hawks use the soft siege strategy with prey energy to complete the hunt, as shown in Eq. (8) and Eq. (9):

$$X(t+1) = \Delta X(t) - E|J \cdot X_{\mathrm{p}}(t) - X(t)| \tag{8}$$

$$\Delta X(t) = X_{\mathrm{p}}(t) - X(t) \tag{9}$$

where, $J = 2(1 - r_5)$ is the prey's movement distance in the jump mode, and r_5 is a random number in $(0, 1)$.

b) Hard besiege: When $r \geq 0.5$ and $|E| < 0.5$, the prey has insufficient energy, and Harris hawks adopt the hard siege strategy for rapid predation, as in Eq. (10):

$$X(t+1) = X_{\mathrm{p}}(t) - E|\Delta X(t)| \tag{10}$$

c) Soft besiege with progressive rapid dives: When $r < 0.5$ and $|E| \geq 0.5$, the prey has sufficient energy to escape. Harris hawks then choose the soft siege combined with a progressive dive tactic, as described by Eq. (11). This strategy includes two hunting methods, with the second selected if the first fails:

$$X(t+1) = \begin{cases} Y : X_{\mathrm{p}}(t) - E|JX_{\mathrm{p}}(t) - X(t)| \\ \quad \text{if } f(Y) < f(X(t)) \\ Z : Y + S \times LF(D) \\ \quad \text{if } f(Z) < f(X(t)) \end{cases} \tag{11}$$

where, D is the spatial dimension, S is a $1 \times D$ random vector, f is the fitness function, and LF is the Levy function simulating the prey's jumping behavior.

d) Hard besiege with progressive rapid dives: When $r < 0.5$ and $|E| < 0.5$, the prey lacks energy but has a chance to escape. Harris hawks then employ the hard siege strategy with a progressive dive to reduce the distance to the prey and form an encirclement, as in Eq. (12):

$$X(t+1) = \begin{cases} Y : X_{\mathrm{p}}(t) - E|JX_{\mathrm{p}}(t) - X_{\mathrm{m}}(t)| \\ \quad \text{if } f(Y) < f(X(t)) \\ Z : Y + S \times LF(D) \\ \quad \text{if } f(Z) < f(X(t)) \end{cases} \tag{12}$$

IV. Problem Model

In the feature selection and parameter optimization model of this study, the decision variables revolve around ReliefF feature screening, performance adjustment of the SVM classifier, and determination of the final feature subset. That is, by adjusting variables such as k, SN, C, γ and FS, the value of the objective function f is minimized, thereby achieving a balance between "high classification accuracy" and "small scale of the feature subset". Therefore, the expression of the optimization problem model in this study is [see Eq. (13)]:

$$\text{Min } f(k, SN, C,\ \gamma,\ FS) \tag{13}$$

The decision variables in the optimization problem model include: ReliefF feature screening parameters (k, the number of nearest-neighbor samples, which is used to determine the scale of neighbor samples referenced when calculating feature weights; SN, the scale of the feature subset selected in the initial screening, i.e., the number of features preliminarily screened out), parameters of the SVM classifier (C, the penalty parameter, which is used to control the penalty intensity for misclassified samples; γ, the kernel function parameter, which defines the mapping complexity of the kernel function in the sample space), and feature selection results (discrete variable FS). The value ranges of each decision variable must satisfy the following constraints [see Eq. (14) to Eq. (18)]:

$$k_{\min} < k = \lfloor \vec{p} \cdot k_{\max} + 0.5 \rfloor < k_{\max} \tag{14}$$

$$SN_{\min} < SN = \lfloor \vec{p} \cdot D + 0.5 \rfloor < SN_{\max} \tag{15}$$

$$C_{\min} < C < C_{\max} \tag{16}$$

$$\gamma_{\min} < \gamma < \gamma_{\max} \tag{17}$$

$$FS_i = \begin{cases} 1 & \text{if the } i\text{th feature is selected} \\ 0 & \text{otherwise} \end{cases} \tag{18}$$

specifically, for the ReliefF parameters: $k_{\min} = 0$, $k_{\max} = 10$; $\text{SN}_{\min} = 0$, $\text{SN}_{\max} = 0.005 \cdot D$. For the SVM classifier parameters, $C_{\min} = 0$, $C_{\max} = 1000$; $\gamma_{\min} = 0$, $\gamma_{\max} = 1$. The feature selection result is characterized by the discrete variable FS, where $\text{FS}_i = 1$ indicates that the i-th feature is selected, and $\text{FS}_i = 0$ indicates that the feature is excluded. The constructed "0/1 binary vector" is used to describe the final feature subset, whose dimension is consistent with the number of original features.

The problem model in this study is oriented towards minimizing the objective function f. The objective function and the expression of the optimization goal are as follows [see Eq. (19) and Eq. (20)]:

$$f = \alpha \cdot (1 - acc) + (1 - \alpha) \cdot \frac{SF}{D} \tag{19}$$

$$acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{20}$$

where, α is the balance coefficient ($\alpha = 0.7$), SF is the number of selected features, D is the total number of original features, and acc represents the classification accuracy. The calculation equation is shown in Eq. (20). The model in this study synchronously optimizes continuous and discrete variables, including ReliefF parameters (k, SN), SVM classifier parameters (C, γ), and feature selection results (discrete FS vector). By leveraging the "synergistic effect" between feature subsets and classifier parameters, it more accurately excavates the "classification discriminant information" of the dataset. This overcomes the problems of traditional univariate optimization methods, such as being prone to falling into local optima and the "curse of dimensionality", where computational complexity grows exponentially with the dimension of variables. Ultimately, it achieves the global optimization of classifier performance.

V. IBHHO for Feature Selection

A. Solution Structure

The solution structure of the algorithm in this study is represented by a coordinate array, where each element of the array is a continuous variable with a value range between $[0, 1]$. The solution structure consists of three parts: the parameters k and SN of ReliefF, the parameters C and γ of the SVM classifier, and the feature selection results. Fig. 1 shows a schematic diagram of the solution structure of the algorithm in this study.

For the first part, the parameters k and SN of ReliefF are calculated by Eq. (2) and Eq. (3), with maximum values set to 10 and 0.005, respectively. For the second part, the parameters C and γ of SVM are set to range between $(0, 1000)$ and $(0, 1)$, respectively. The third part represents the result of discrete feature selection. Given that feature selection is inherently discrete and HHO is designed for continuous optimization problems, a discrete version of HHO (BHHO) must be employed. In the BHHO algorithm, the result vector of feature selection is represented by Eq. (21):

$$FS_i = \begin{cases} 1 & \text{if } w_i < g_{i+4} \\ 0 & \text{otherwise} \end{cases} \tag{21}$$

where, w_i is a random number within the interval $(0, 1)$, g_{i+4} representing the $i+4$-th position in the solution structure.

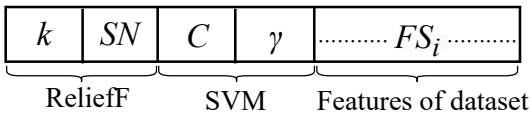


Fig. 1. Solution structure.

B. Fitness function

The fitness function is used to measure the quality of each Harris hawk individual in the IBHHO algorithm. Each Harris hawk represents a potential feature subset along with parameters of the classifier and ReliefF, and the fitness function provides an evaluation criterion for these individuals. The algorithm uses this criterion to select and evolve toward better feature subsets and classifier parameters. Feature selection is a multi-objective optimization problem that should simultaneously consider higher classification accuracy and fewer features. Linear weighting methods can transform multiple objectives into a single fitness value to facilitate algorithmic solution. In this study, a linear combination of classification accuracy and the number of selected features is used as the fitness function, as defined in Eq. (19).

C. Co-Optimization Strategy for Parameters and Features Based on HHO Two-Stage Characteristics

As described in the first subsection, the solution structure of IBHHO is divided into three parts (ReliefF, classifier parameters, and feature selection results). Since the HHO algorithm has both exploration and exploitation components, it effectively balances global search and local search.

The proposed staged optimization strategy fully capitalizes on the complementary characteristics of the HHO algorithm's exploration and exploitation phases:

During the exploration phase, the search space is optimized to identify potential high-quality solution regions. Therefore, ReliefF and SVM parameters are optimized during this phase. By dynamically adjusting the ReliefF parameters, the importance of features is evaluated from multiple perspectives, generating stable feature rankings. Concurrently, SVM parameters are adjusted to rapidly adapt to the data distribution, thereby enhancing model performance;

The development phase focuses on fine-grained search around high-quality solutions. At this stage, feature subset and SVM parameter optimization are performed. Based on the reliable feature subset generated by ReliefF parameters in the exploration phase, the focus shifts to fine-tuning SVM parameters and further controlling the feature subset size,

achieving a balance between computational efficiency and optimization accuracy.

This design avoids the computational bottleneck of direct feature selection in high-dimensional data while reducing the search space complexity through separate parameter optimization, providing a new optimization form for the hybrid feature selection framework.

D. Elite Feature Guidance Strategy

In the feature selection process of the original HHO algorithm, the optimization of feature subsets is simulated as the process of hawk flocks chasing prey. The candidate feature subsets correspond to the positions of hawks, while the optimal feature subset is analogized to the prey's position. The algorithm continuously adjusts the positions of hawks through two stages—exploration and exploitation—to approach the prey. During the exploitation phase, hawks mainly update their positions using Eq. (8), Eq. (10), Eq. (11) and Eq. (12), to attempt getting closer to the prey. However, this updating approach in high-dimensional feature spaces may lead to extensive redundant searches in non-critical regions. Moreover, the lack of a protection mechanism for verified high-quality features makes the algorithm prone to missing the true optimal feature subset, resulting in slow convergence and limited result accuracy.

The Elite Feature Guidance Strategy proposed in this study improves the exploitation phase of the original HHO algorithm by introducing pre-identified key features (marked as 1) as fixed "anchor points", significantly optimizing the feature selection optimization process. Specifically, the algorithm directly assigns the "prey positions" of these key features to strict binary value 1, keeping them unchanged throughout the search process. These anchored features act as navigation coordinates, guiding the hawk flock to converge preferentially toward the optimal feature subset containing these key features. During the iterative update process, the algorithm first ensures that the anchored features in the candidate feature subset always remain 1, avoiding the loss of these key features during the search. Eq. (22) shows the position update method after adding the guidance mechanism. Among them, X_{new} is the new prey position (candidate solution), and i is the position in the solution structure array.

When $i < 4$, it represents the first 4 positions of the solution structure, i.e., the ReliefF and SVM parameters, which do not participate in the elite guidance mechanism; when $i > 4$, it represents the position of the feature selection result, which is the part involved in the elite guidance.

$$X'_{\text{p}}(t)_i = \begin{cases} X_{\text{p}}(t)_i & (i \in [0,3]) \\ FS_{i+4} & (i \geq 4) \end{cases} \tag{22}$$

For non-anchored features, during the exploitation phase, the optimization of the feature subset in the new position of the hawk is updated using Eq. (23), Eq. (24), Eq. (25) and Eq. (26), where X'_{rabbit} represents the position of prey containing fixed anchoring features and $X_{\text{new}}(t+1)$ is the updated candidate feature subset.

$$\begin{cases} X_{\text{new}}(t+1) = \Delta' X(t) - E|J \cdot X'_{\text{p}}(t) - X(t)| \\ \Delta' X(t) = X'_{\text{p}}(t) - X(t) \end{cases} \tag{23}$$

$$X(t+1) = X'_{\text{p}}(t) - E|\Delta' X(t)| \tag{24}$$

$$X_{\text{new}}(t+1) = \begin{cases} Y : X'_{\text{p}}(t) - E|J \cdot X'_{\text{p}}(t) - X(t)| \\ \quad f(Y) < f(X(t)) \\ Z : Y + S \times LF(D) \\ \quad f(Z) < f(X(t)) \end{cases} \tag{25}$$

$$X_{\text{new}}(t+1) = \begin{cases} Y : X'_{\text{p}}(t) - E|J \cdot X'_{\text{p}}(t) - X_{\text{m}}(t)| \\ \quad f(Y) < f(X(t)) \\ Z : Y + S \times LF(D) \\ \quad f(Z) < f(X(t)) \end{cases} \tag{26}$$

In this way, the algorithm always builds upon anchored features when exploring new feature combinations, effectively reducing search attempts in non-critical regions. By fixing key features, this strategy significantly narrows the effective search space and minimizes redundant exploration, thereby enhancing the algorithm's convergence speed.

Meanwhile, since anchored features are pre-verified high-quality features, the algorithm can more accurately approach the true optimal feature subset, ultimately improving the accuracy and efficiency of feature selection–particularly suitable for feature selection tasks on high-dimensional complex datasets.

E. Algorithm Flow of IBHHO

In summary, addressing the challenge of feature selection for high-dimensional microarray datasets, this study proposes three innovative strategies. Firstly, a hybrid feature selection framework integrating filter (ReliefF) and wrapper (SVM) methods is constructed to synchronously optimize ReliefF parameters, SVM hyperparameters, and feature subset size. The feature subset size parameter is adaptively adjusted based on the intrinsic dimensionality of the dataset, enabling dynamic adaptation. Secondly, a differentiated optimization strategy is designed based on the two-stage characteristics of the HHO algorithm: the exploration phase optimizes ReliefF and SVM parameters to comprehensively search the feature space, while the exploitation phase focuses on feature subset refinement and classifier parameter fine-tuning, significantly enhancing classification performance. Finally, an "Elite Feature Guidance Strategy" is proposed, which fixes pre-identified key features as binary value 1 as "anchor points" in the search space. This guides the algorithm to converge rapidly to the optimal feature subset, effectively reducing redundant exploration and substantially improving the efficiency and accuracy of feature selection. The flowchart of the proposed IBHHO algorithm is shown in Fig. 2.

Initialization Phase: The basic parameters of the algorithm (population size N and maximum number of iterations T). Then initialize the positions of hawks, including ReliefF parameters (k and SN), SVM parameters (C and γ), and feature

subsets (where binary values indicate whether features are selected).

Fitness Calculation: The fitness is computed using Eq. (19). For the current hawk positions (candidate solutions), feature selection is performed on the microarray dataset based on the current ReliefF parameters to obtain the feature subset SN for SVM training. The selected feature subset SN is used to train the SVM and further refine the optimal subset, after which the classification accuracy acc is calculated. Finally, the fitness value is computed using Eq. (19). A smaller fitness value indicates better comprehensive performance of the candidate solution (higher classification accuracy with fewer features).

Exploration Phase: Execute "broad-spectrum optimization of ReliefF and SVM parameters". The core objective is to simulate the behavior of hawk flocks searching for prey in a vast space using the HHO algorithm, performing extensive and diverse adjustments to ReliefF feature evaluation parameters and SVM classifier hyperparameters. This aims to identify potential high-quality parameter intervals for the subsequent exploitation phase.

Exploitation Phase: Fine-tune SVM parameters and refine feature subsets. In this phase, the Elite Feature Guidance Mechanism is introduced, where features marked as 1 (pre-identified key features) have their "prey positions" directly assigned as strict binary value 1. The positions of hawks are updated using Eq. (23), Eq. (24), Eq. (25) and Eq. (26), guiding the algorithm to converge rapidly to the verified optimal feature subset.

Iteration Termination Criterion ($t < T$?): If $t < T$, the process returns to the fitness calculation phase for the next round of optimization; if $t > T$, the iteration terminates and the optimal solution is output.

VI. Results and Discussion

In this section, we evaluate the performance of the proposed method in feature selection tasks for medical datasets through comparative analysis from different aspects. All experiments were conducted on the MATLAB 2023b platform, with the hardware configuration as follows: Windows 11 operating system, Intel (R) Core (TM) i5-13500H CPU @ 2.6 GHz, and 16.0 GB of RAM.

A. Description of Datasets

To analyze the performance of the proposed method, we utilized eight high-dimensional DNA microarray gene expression datasets, including both binary and multi-class classifications. These datasets were downloaded from two public websites: http://case.szu.cn/stff/zhuzx/Datssets.html and https://github.com/kivancguckiran/microarray-data. Table I provides the specific descriptions of the datasets.

Overall, the 8 microarray datasets selected in this study are distinctly diverse and representative, and they have also been used in existing literature, such as references [30], [31], and [24]. These datasets can comprehensively and rigorously verify the performance of the proposed method: in terms of classification tasks, they include both binary classification (such as Colon Cancer, CNS, Leukemia, Lung cancer, Ovarian cancer, Breast cancer) and multi-class classification (SRBCT with 4 classes, Burcyznski with 3 classes), which can test the adaptability of the method to different category complexities; in terms of sample composition, the total number of samples ranges from 60 (CNS) to 253 (Ovarian); the dimensionality of disease features is distributed in a gradient from 2000 to 22,283, which fully verifies the generalization ability of this method in mining key features and improving classification accuracy. The selection of these datasets provides a comprehensive experimental basis for evaluating the effectiveness and reliability of the method.

TABLE I. Datasets Descriptions

Dataset	Samples	Features	Classification type
Colon Cancer	62	2000	Binary-class
SRBCT	63	2308	Multi-class
Central Nervous System (CNS)	60	7129	Binary-class
Leukemia	72	7129	Binary-class
Lung cancer	181	12533	Binary-class
Ovarian cancer	253	15154	Binary-class
Breast cancer	118	22215	Binary-class
Burcyznski	127	22283	Multi-class

B. Experimental Design and Performance Indicators

To evaluate the performance of machine learning models on different datasets, this study employed k-fold cross-validation. The datasets were further randomly divided into training sets and independent test sets via k-fold cross-validation. The main advantage of cross-validation lies in the independence of each test set, thereby enhancing the reliability of the results. In this study, the value of k was set to five. Thus, the datasets were partitioned into five parts, with each part containing data in equal proportion from each class. Four parts were used for training, while the remaining one part served as the test set.

The analysis of feature selection results for the eight datasets employed multiple evaluation metrics, including accuracy, fitness value, number of selected features, and execution time. These statistical metrics aim to analyze experimental outcomes and determine the algorithm's performance.

To robustly evaluate the reliability of the proposed method, this study is designed to compare all performance indicators of the proposed algorithm with those of all comparative algorithms in the experimental section, and the finally reported results are the average values of 30 independent executions.

C. Overall Performance Evaluation

This section presents a comparison of the IBHHO algorithm with other optimization algorithms in terms of evaluation metrics including classification fitness, accuracy, number of selected features, and execution time. It is divided into two parts: comparison of IBHHO with various optimization algorithms aided by ReliefF, and comparison of IBHHO with various state-of-the-art methods.

1) Comparison of IBHHO with various optimization algorithms aided by ReliefF: In this subsection, four optimization algorithms integrated with ReliefF are employed as comparative algorithms, namely GA [25], PSO [32], FA [33], and WCA [34].

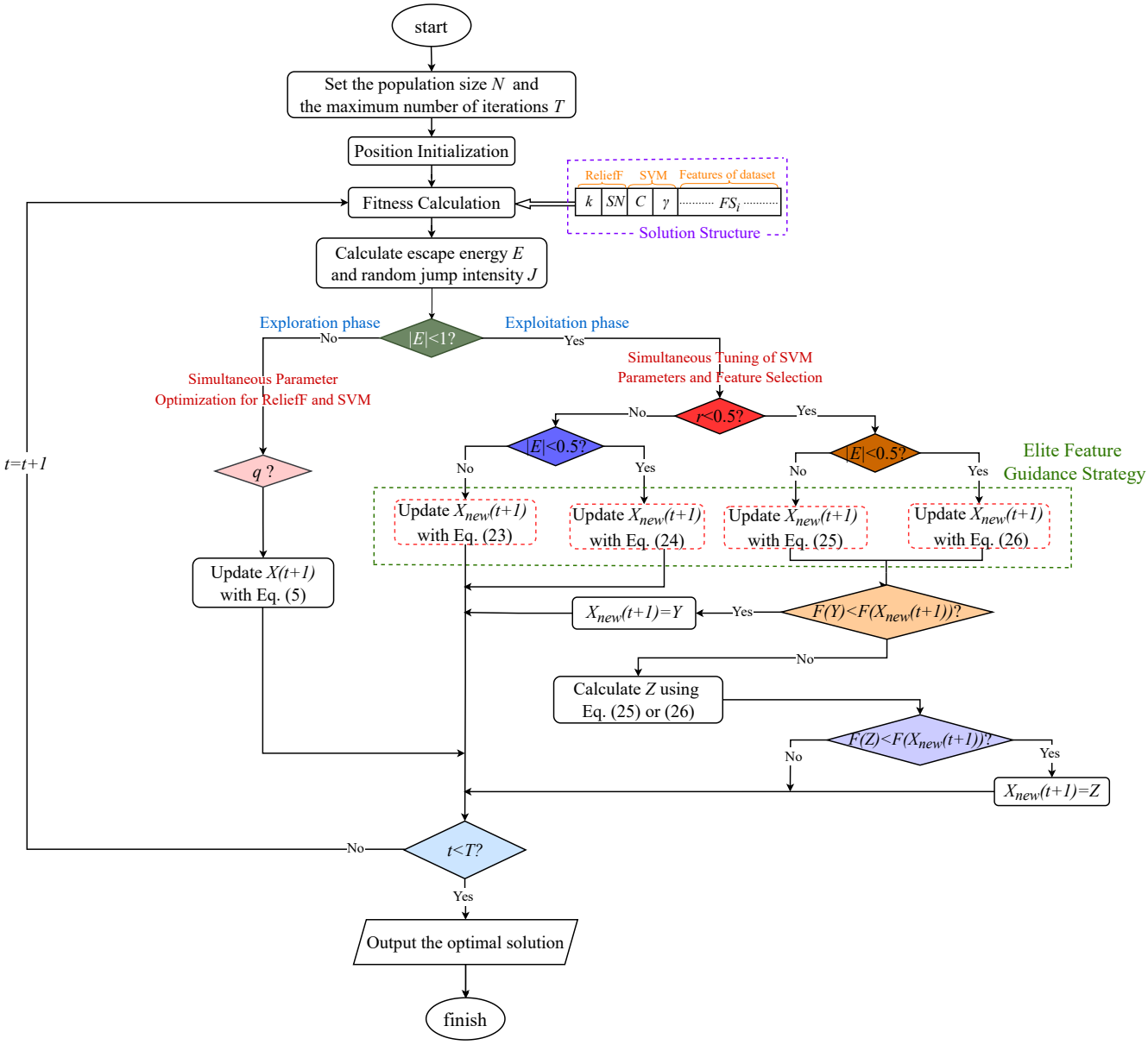


Fig. 2. The flowchart of the IBHHO.

Boxplot analysis is the optimal method to represent the data distribution characteristics of the collected results and identify data anomalies (such as skewness and outliers). A boxplot displays data distribution in the form of different quartiles, specifically the lower (the lowest point/edge of the whisker) and upper (the highest point/edge of the whisker) quartiles, which represent the minimum and maximum values of the data distribution. The lower quartile and upper quartile are indicated by the corners of the rectangle. A small boxplot rectangle indicates a stronger consistency of the data.

To verify the performance of the IBHHO algorithm, four algorithms integrated with ReliefF (GA-R, PSO-R, FA-R, WCA-R) were selected as comparative algorithms. Experiments were conducted on 8 datasets, with fitness as the evaluation metric (a smaller fitness value indicates better optimization performance of the algorithm). The boxplot results are shown in Fig. 3.

On the Colon Cancer dataset, it is intuitively visible that the box of the IBHHO algorithm is the narrowest, which clearly indicates that the fluctuation range of its multiple running results is the smallest, and its stability is the best among the five algorithms. In contrast, the GA+R algorithm has the widest box and the worst stability. The small squares inside the boxes represent the average values of the algorithm runs. By comparison, the small square corresponding to IBHHO is located at the lowest position, which strongly proves that the average value of its fitness is the smallest and its classification performance is more excellent. The performances of the PSO+R, FA+R, and WCA+R algorithms are relatively close. Their box widths are between those of IBHHO and GA+R, with moderate stability, and their mean points are concentrated in a specific interval, with average fitness better than that of GA+R.

On the SRBCT, Lung cancer, and Ovarian cancer datasets,

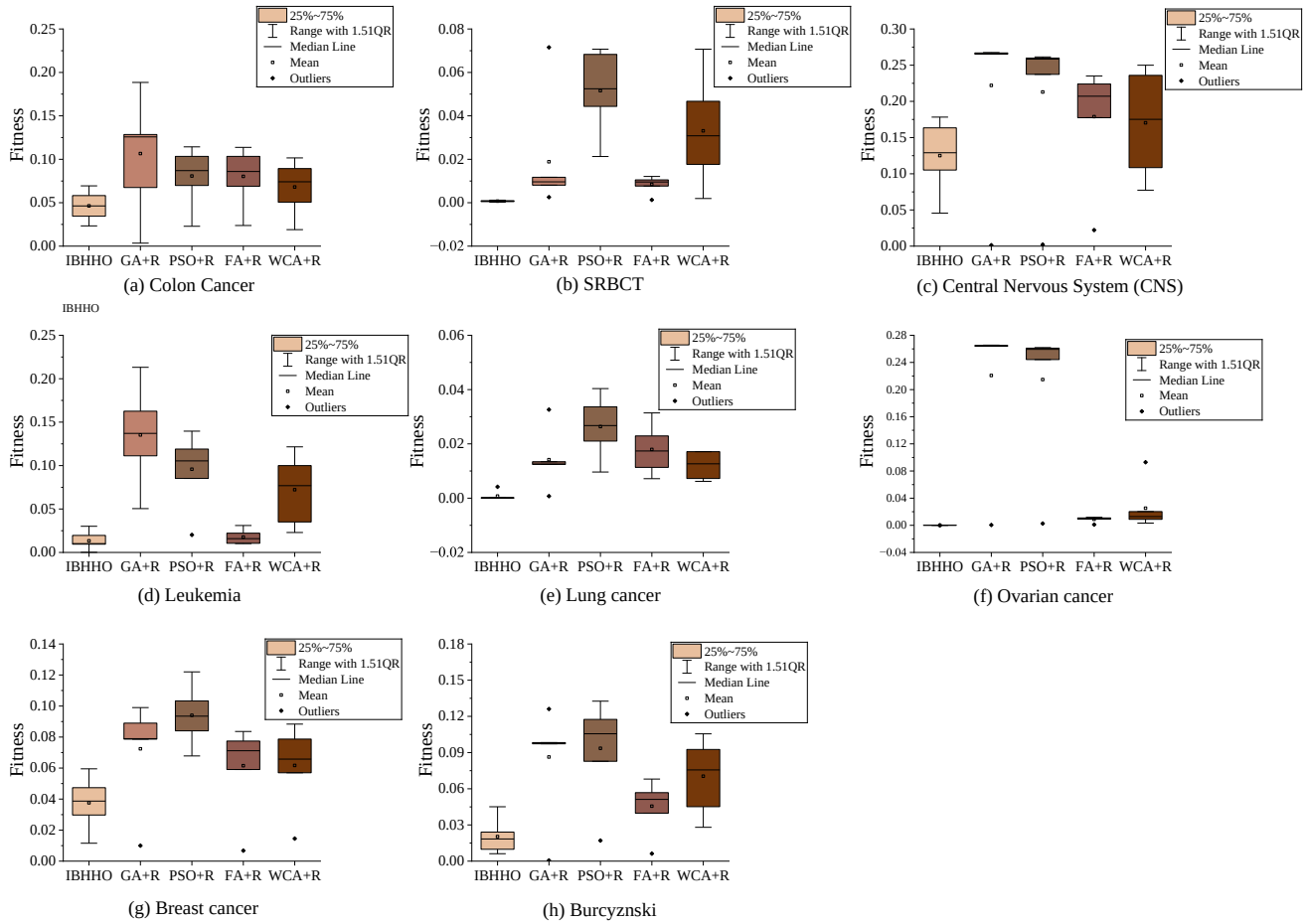


Fig. 3. Comparison of average fitness among different feature selection optimization algorithms.

the box of the IBHHO algorithm appears as a thin line. Compared with the other four algorithms, its box has the smallest width, indicating that the results of multiple runs are highly convergent and its stability is optimal. When comparing the small squares representing the average values of each algorithm, the small square corresponding to the IBHHO algorithm has the lowest fitness, demonstrating its best classification performance.

On the Central Nervous System (CNS), Breast cancer, and Burcyznski datasets, the box of the GA+R algorithm is the narrowest, showing the best stability, while the stability of the IBHHO algorithm is relatively inferior on these three datasets. However, when comparing the small squares inside the boxes that represent the average fitness values, the small square of IBHHO is located at the lowest position, whereas that of GA+R is at the highest. This proves that the average fitness of the IBHHO algorithm is the lowest, indicating better fitness performance, while the fitness of the GA+R algorithm is the worst.

From the boxplots of all test datasets, the key statistical metrics such as the median and mean of the IBHHO algorithm's boxplot are significantly lower than those of the comparison algorithms, and the overall distribution of the box is more concentrated in the low fitness interval. This fully indicates that, relying on its unique population update and search strategies, IBHHO can more efficiently traverse the search space and explore low-fitness solutions in different data scenarios. Its optimization stability and effectiveness are superior to those of the comparison algorithms, which strongly verifies that IBHHO exhibits excellent performance across multiple types of datasets.

To verify the comprehensive performance of the IBHHO algorithm in feature selection tasks, experiments were conducted on eight datasets using the proposed algorithm and four comparative algorithms with ReliefF assistance, as shown in Fig. 4. The classification performance was measured by the average classification accuracy of 30 independent runs, and the feature screening efficiency was evaluated by the average number of selected features. The results were visualized through a combination of bar charts and line charts (the purple bars represent the average accuracy, and the red lines represent the average number of selected features).

As shown in the figure, on the four datasets of Colon Cancer, Central Nervous System (CNS), Breast cancer, and Burcyznski, the accuracy of IBHHO is higher than that of the other four comparative algorithms. Moreover, the number of selected features by the IBHHO algorithm is the smallest. This demonstrates that on these four datasets, IBHHO can ensure higher precision while screening key features more efficiently, well-verifying the advantages of the IBHHO algorithm.

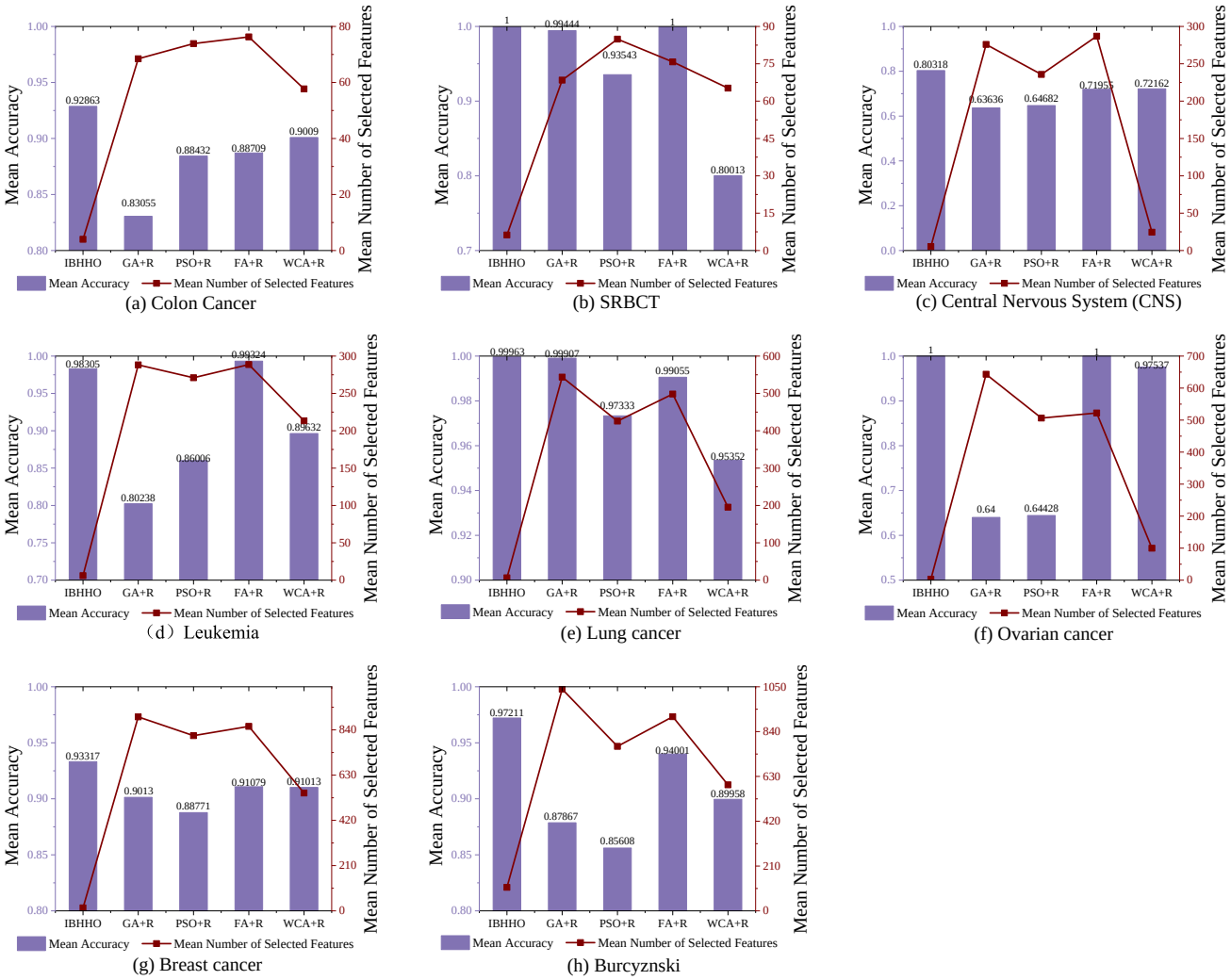


Fig. 4. Comparison of the number of features and accuracy.

On the SRBCT and Ovarian cancer datasets, the accuracy of IBHHO is on par with that of the FA algorithm, while the accuracies of the GA+R algorithm and the WCA+R algorithm are slightly lower than those of IBHHO and FA+R, indicating that these three algorithms have advantages in classification accuracy on the SRBCT dataset. However, considering the number of selected features, on the SRBCT dataset, the number of selected features by the IBHHO algorithm (6) is much lower than that by the GA+R and FA+R algorithms (68 and 75 respectively); on the Ovarian cancer dataset, the number of selected features by IBHHO is 2, compared to 522 and 99 by FA+R and WCA+R. This proves that IBHHO can screen key features while ensuring high precision, and performs better in the "classification accuracy-feature efficiency" dimension.

On the Leukemia dataset, the FA+R algorithm has the highest classification accuracy (0.99), with IBHHO being slightly less accurate (also 0.99). However, the number of selected features by IBHHO is 6, while that by the FA+R algorithm reaches 288. This shows that the ability of IBHHO to screen key features is far superior to that of FA+R. Therefore, it can be concluded that IBHHO performs better in meeting the dual requirements of "accuracy+number of selected features".

On the Lung cancer dataset, all algorithms exhibit classification performance with accuracy close to 1. Nevertheless, the number of selected features by the IBHHO algorithm is much lower than that of the comparative algorithms, which proves that the IBHHO algorithm can select the most critical features while ensuring high accuracy.

From the perspective of meeting the dual requirements of "accuracy+number of selected features" across all test datasets, compared with comparative algorithms such as GA+R, PSO+R, FA+R, and WCA+R, the IBHHO algorithm, relying on its unique search and update mechanisms, achieves efficient screening of key features while ensuring high accuracy. This not only reduces model complexity but also verifies the advantages of the IBHHO algorithm.

Table II presents the comparison of the average execution times of the proposed algorithm and comparative algorithms over 30 independent runs under the same operating environment.

As shown in the table, on the two low-dimensional datasets of Colon Cancer (62 samples, 2000 genes) and SRBCT (63 samples, 2308 genes), GA+R achieves the shortest execution

time. This is because in low-dimensional spaces, its genetic operations can converge directly, and it rapidly screens features through simple operations such as selection and crossover, thereby reducing redundancies. PSO+R has a slightly longer execution time, which stems from the fact that its particle update rules require more iterations to adjust particle distribution during the initial exploration of small-scale data. IBHHO ranks third with an execution time close to that of PSO+R, mainly because its unique search mechanism involves fine-grained exploration of the solution space during initialization or early iterations, resulting in additional computations and thus a slightly longer execution time.

On two medium-high dimensional datasets, Central Nervous System (CNS) (60 samples, 7129 genes) and Leukemia (72 samples, 7129 genes), PSO+R exhibits the shortest execution time. This is because its particles can rapidly focus on key features in high-dimensional spaces by sharing global optimal information, thus avoiding the computational bottlenecks of GA+R's high-dimensional crossover operations and the exploratory redundancy of IBHHO. IBHHO ranks second with an execution time comparable to those of GA+R and PSO+R, as its strategy to escape local optima during high-dimensional searches introduces additional computational overhead.

On the Lung Cancer dataset (181 samples, 12,533 genes), GA+R runs the fastest. The larger sample size strengthens the selection pressure of GA+R, enhancing its efficiency in eliminating redundant genes and achieving directional evolution in high-dimensional spaces. However, for IBHHO, under high-dimensional and large-sample scenarios, the high computational cost associated with its complex feature screening strategy results in it being slower than both GA+R and PSO+R.

On ultra-large-scale high-dimensional datasets such as Ovarian cancer (253 samples, 15,154 genes), Breast cancer (118 samples, 22,215 genes), and Burcyznski (127 samples, 22,283 genes), IBHHO achieves the shortest running time. Its heuristic search or group-based screening mechanisms can directly skip redundant features, overcoming the computational breakdown issues of GA+R and PSO+R in ultra-high-dimensional spaces, and is particularly adaptable to extreme data scenarios involving multi-class classification and ultra-large-scale genes. In these cases, IBHHO ranks first, while GA+R and PSO+R exhibit significantly longer running times due to the high-dimensional computational complexity and inefficiency in handling multi-class classification tasks, which are much longer than that of IBHHO.

In summary, IBHHO is not always the fastest; instead, its performance varies dynamically with data dimensionality: it is outperformed by GA+R in low-dimensional cases (as GA+R features more straightforward computations), and by PSO+R in medium-to-high-dimensional scenarios (since PSO+R achieves more efficient convergence in high dimensions). However, in ultra-large-scale high-dimensional scenarios, it emerges as the optimal choice by virtue of its unique mechanisms, reflecting its targeted adaptation to different data characteristics. Moreover, on all datasets, IBHHO demonstrates superior classification performance compared to the comparative algorithms.

2) Comparison between proposed algorithm and various state-of-the-art methods: To verify the performance of the IBHHO algorithm, four advanced optimization algorithms, namely AltWOA [20], SAGA [35], AIEOU [36], and BSNDO [37], were selected as comparative algorithms. Experiments were conducted on 8 datasets, with all experiments independently executed 30 times under the same environment.

Fig. 5 presents the box plots of the average fitness values obtained from 30 independent runs. On the Colon Cancer dataset, the box of the SAGA algorithm appears as a thin line, indicating a high degree of convergence (minimal fluctuation) among the results of multiple runs and thus demonstrating the best stability among all algorithms. When comparing the mean points (small squares) within the boxes, IBHHO exhibits a significantly lower average fitness value than the other comparative algorithms, indicating the best classification performance. Moreover, the lowest position of IBHHO's box also confirms its lowest fitness value and optimal classification performance.

On the SRBCT, Lung cancer, and Ovarian cancer datasets, the box of the IBHHO algorithm appears as a thin line with the lowest position among all boxes. The small square within its box (representing the mean fitness value) is also lower than those of other comparative algorithms, which proves that IBHHO achieves the best classification performance. Following that, the performances of the SAGA and AltWOA algorithms are second to IBHHO, while the BSNDO algorithm exhibits the worst performance.

On the Central Nervous System (CNS), Breast cancer, and Burcyznski datasets, the SAGA algorithm exhibits the narrowest box, indicating the smallest fluctuation in results across multiple runs and the best stability among all algorithms. However, comparing the mean points (small squares) within the boxes, IBHHO's mean point is positioned lowest, meaning its average fitness value is far superior to SAGA (whose mean point is higher, reflecting the worst fitness performance). This shows that although IBHHO is slightly less stable than SAGA (with a marginally wider box), it achieves a breakthrough in fitness optimization through its unique search strategy. It compensates for the stability difference with a lower average fitness (better classification performance), demonstrating strong adaptability to high-dimensional complex data.

On the Leukemia dataset, the box width of IBHHO is close to that of SAGA (moderate stability), but its mean point is the lowest, showing a significant advantage in fitness. Compared with algorithms like AltWOA, AIEOU and BSNDO, IBHHO is superior in both stability and fitness. Its box is concentrated in the low-fitness interval with short whiskers (data is concentrated without extreme fluctuations), verifying its efficient optimization ability on medium-dimensional data.

Boxplots across all datasets show that statistical metrics such as the median and mean of IBHHO all lie in the lowest interval; its box is overall concentrated in the low-fitness range with short whiskers (indicating high aggregation of non-outlier data). This indicates that through its unique population update and search mechanisms, IBHHO can efficiently traverse the solution space and explore low-fitness solutions across different data scenarios (low-dimensional, medium-to-high-dimensional, and ultra-large-scale high-dimensional), avoiding the drawbacks of comparative algorithms (such as the high fluctuation of SAGA and the local convergence bias of AltWOA). Even in datasets where SAGA dominates in

TABLE II. COMPARISON OF AVERAGE EXECUTION TIME(S)

Dataset	IBHHO	GA+R	PSO+R	FA+R	WCA+R
Colon Cancer	76.1	47.8	75.8	342.8	476.5
SRBCT	205.7	179.5	194.7	935.6	1156.7
Central Nervous System (CNS)	107.4	104.0	103.6	1711.5	103.3
Leukemia	115.5	119.7	111.9	936.8	341.6
Lung cancer	207.5	173.5	197.9	793.6	258.8
Ovarian cancer	283.0	365.6	365.3	920.2	361.3
Breast cancer	209.3	221.3	205.1	1025.7	364.0
Burcyznski	325.7	408.6	408.0	1535.3	277.0

stability (CNS), IBHHO still outperforms with a lower average fitness (a core performance metric), fully verifying its excellent generalization ability and optimization efficiency on multi-type microarray data.

To verify the comprehensive performance of the IBHHO algorithm in feature selection tasks, the algorithm proposed in this study and four comparative algorithms from various state-of-the-art methods were tested on 8 datasets, as illustrated in Fig. 6. The classification performance is evaluated using the average classification accuracy from 30 independent runs (represented by blue bar charts), while the feature selection efficiency is depicted by orange curves (showing the average number of selected features). This dual-metric visualization (precision+feature count) precisely assesses the algorithm's balance between "classification accuracy" and "feature selection efficiency".

On the Colon Cancer and SRBCT datasets, IBHHO achieves the best classification accuracy (reaching above 0.9), while other algorithms are around 0.8. Looking at the number of selected features, although GA selects fewer key features (1 and 4, respectively, compared to IBHHO's 4 and 6), its classification accuracy is lower. Overall, IBHHO demonstrates a better balance between "accuracy" and "efficiency". On the Central Nervous System (CNS), Lung cancer, Ovarian cancer, and Breast cancer datasets, the IBHHO algorithm achieves the best classification accuracy, with SAGA being slightly inferior. However, when comparing the number of selected features, the number of key features selected by the IBHHO algorithm is less than that selected by SAGA, which proves that IBHHO can search for more critical features while ensuring high accuracy.

On the Leukemia dataset, IBHHO achieves a classification accuracy of 0.98 with only 6 selected key features, demonstrating the best balance performance between "accuracy and feature selection efficiency". In contrast, the BSNDO algorithm exhibits a classification accuracy of only 0.62 while selecting up to 3519 key features, thus showing the worst performance in balancing "accuracy and number of features". On the Burcyznski dataset, the AltWOA algorithm selects the fewest features (26), while IBHHO selects 110 features. However, in terms of classification accuracy, IBHHO reaches 0.97, whereas AltWOA only achieves 0.52. Therefore, IBHHO still delivers the best comprehensive performance on the Burcyznski dataset.

The above results fully demonstrate IBHHO's advantages in simultaneously maximizing classification accuracy and minimizing the number of features. Through efficient search mechanisms, this method can generate sparse yet high-quality feature sets, reducing model complexity while enhancing performance. This makes IBHHO a powerful tool in the field of bioinformatics—its unique advantages are particularly prominent in research scenarios where both accuracy and feature efficiency are equally emphasized to obtain reliable biological insights (such as identifying the minimal gene panel required for cancer diagnosis).

Table III presents a comparison table of the average execution time from 30 independent runs of the proposed algorithm and the comparative algorithms under the same operating environment. As can be observed from the table, on the Colon cancer (62 samples, 2000 features) and SRBCT (63 samples, 2308 features) datasets, which belong to low-dimensional spaces, the simple search strategy of BSNDO avoids the fine-grained search overhead during the initialization of IBHHO and converges directly, resulting in shorter computation time. For the Central Nervous System (CNS) dataset (60 samples, 7129 features), when the sample size is limited and the feature dimension is relatively high, the heuristic search of IBHHO avoids the blind exploration of the particle swarm of AltWOA in the initial iteration and directly converges to the key features.

On the Leukemia (72 samples, 7129 features), Lung cancer (181 samples, 12,533 features), and Breast cancer (118 samples, 22,215 features) datasets, where the sample size is relatively small but the feature dimension is extremely high, AltWOA exhibits faster running speed. This is likely due to the optimizations for high-dimensional computational efficiency in its improved version (AltWOA), which may adopt feature subspace operations or highly optimized vector computation implementations. These optimizations significantly reduce the constant factor overhead of high-dimensional vector operations in each iteration.

On the Ovarian cancer (15,154 features, 253 samples) and Burcyznski (22,283 features, 127 samples) datasets, where both sample sizes and feature dimensions are higher, IBHHO runs faster. Its advantage may lie in its more complex and adaptive exploration-exploitation balance strategies. These strategies allow IBHHO to more effectively utilize information and accelerate the convergence process in the more complex search spaces induced by increased sample sizes, thereby gaining a significant advantage in the total number of iterations and offsetting the potentially higher computational cost of its single iteration.

To summarize, in low-dimensional spaces, although IBHHO is not the fastest in such scenarios, its fitness values

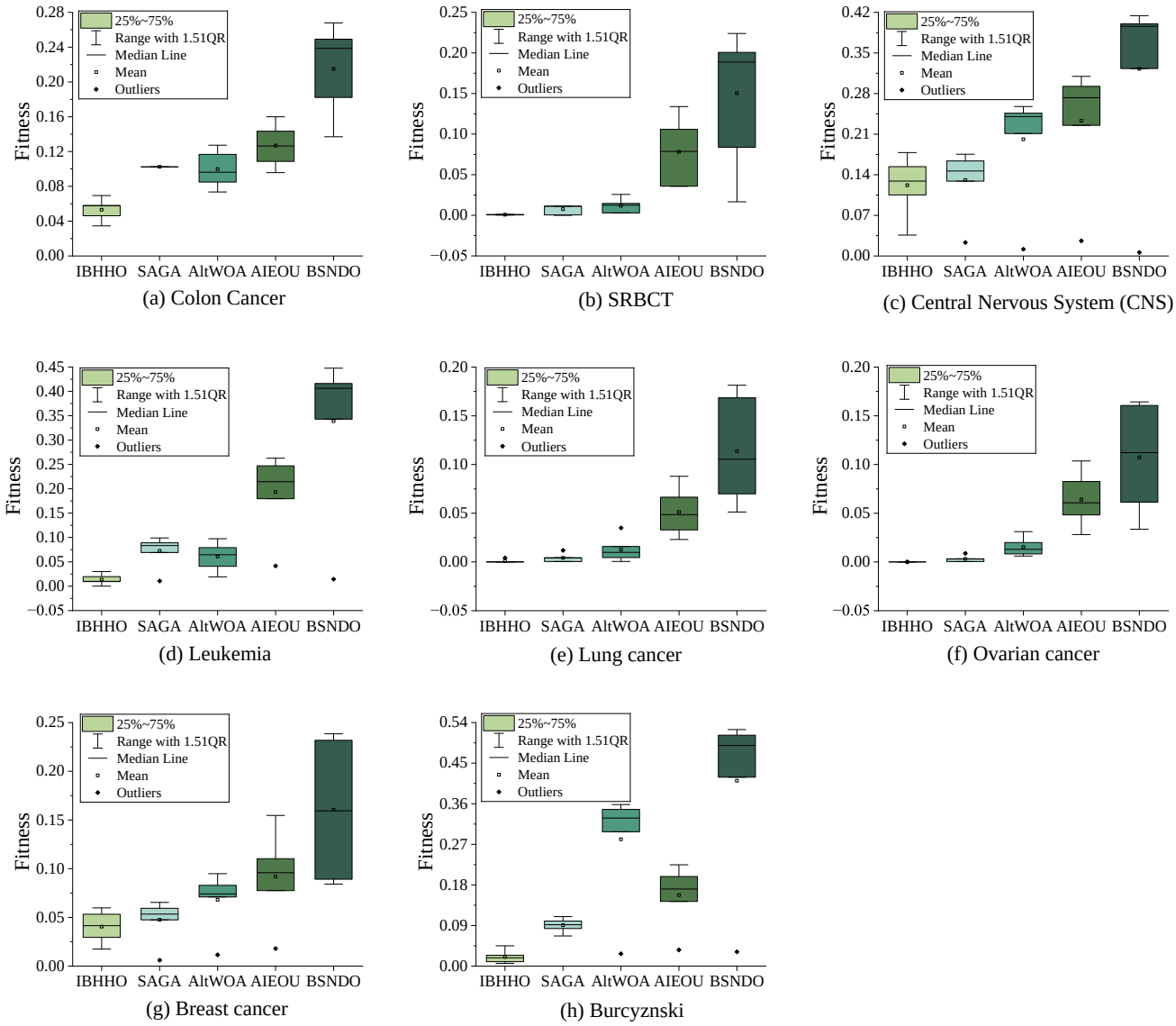


Fig. 5. The box plots of the average fitness values obtained from 30 independent runs.

and classification performance (classification accuracy, number of selected features, etc.) far surpass those of BSNDO, demonstrating decisive advantages. In high-dimensional small-sample scenarios (Leukemia, Lung cancer, and Breast cancer), AltWOA achieves the fastest running speed by significantly reducing the constant factor overhead of each iteration through feature subspace operations and vectorized computation optimizations. However, IBHHO still maintains acceptable timeliness (slightly slower than AltWOA) while comprehensively outperforming AltWOA in both fitness and classification performance. In high-dimensional large-sample datasets (Ovarian cancer and Burczynski), IBHHO relies on adaptive exploration-exploitation strategies to efficiently guide the search direction in complex solution spaces, surpassing AltWOA in speed with fewer iterations while maintaining a leading position in classification performance.

TABLE III. COMPARISON OF AVERAGE EXECUTION TIME(S) FROM 30 INDEPENDENT RUNS

Dataset	IBHHO	SAGA	AltWOA	AIEOU	BSNDO
Colon Cancer	76.1	178.7	65.0	64.9	62.0
SRBCT	205.7	133.3	100.4	266.6	81.1
Central Nervous System (CNS)	107.4	510.5	163.2	299.3	208.9
Leukemia	115.5	464.9	95.5	204.0	201.8
Lung cancer	207.5	3262.9	68.1	648.7	528.7
Ovarian cancer	283.0	3382.5	376.3	799.2	894.7
Breast cancer	209.3	12832.0	93.6	839.2	735.4
Burcyznski	325.7	11428.3	501.5	827.3	714.6

VII. CONCLUSION

This study aims to address the challenge of feature selection in high-dimensional microarray datasets, where redundant

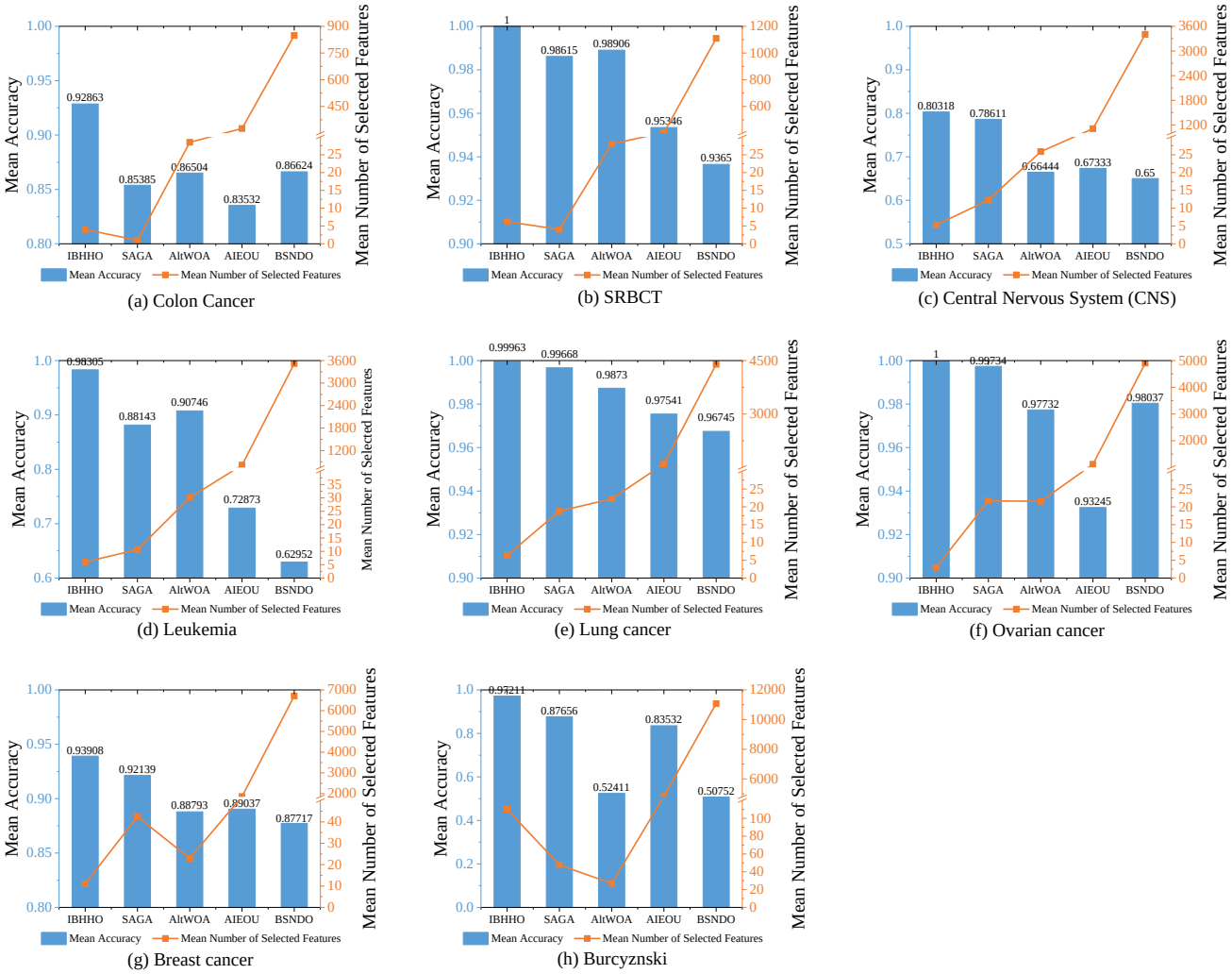


Fig. 6. Comparison chart of the number of features and accuracy.

features and the "curse of dimensionality" severely hinder the performance of machine learning models. To tackle this issue, IBHHO is proposed, which integrates a hybrid feature selection framework, a two-stage parameter-feature co-optimization strategy, and an elite feature guidance mechanism. By fusing filter (ReliefF) and wrapper (SVM) methods, this framework enables the simultaneous optimization of ReliefF parameters, SVM hyperparameters, and feature subsets. Leveraging the two-stage characteristics of HHO (exploration and exploitation), the algorithm focuses on optimizing ReliefF and SVM parameters during the exploration phase, while refining feature subsets in the exploitation phase, thus achieving a balance between global search and local optimization. The elite feature strategy further accelerates the algorithm's convergence speed and reduces redundant exploration by retaining key features as fixed anchors. These innovations collectively enhance IBHHO's ability to identify optimal feature subsets, not only improving classification accuracy but also effectively reducing feature dimensionality. Experimental results demonstrate that IBHHO outperforms other comparative algorithms in terms of comprehensive performance across eight microarray datasets, verifying its effectiveness in handling high-dimensional biological data. In future, it will also be possible to combine bioinformatics knowledge to conduct in-depth analyses of the biological relevance of feature subsets selected by IBHHO, providing more specific guidance for research on cancer gene functions.

REFERENCES

[1] N. Altman and M. Krzywinski, "The curse(s) of dimensionality," Nature Methods, vol. 15, no. 6, pp. 399-400, 2018.

[2] F. Alharbi and A. a.-O. Vakanski, "Machine Learning Methods for Cancer Classification Using Gene Expression Data: A Review," Bioengineering (Basel, Switzerland), vol. 10, no. 2, p. 173, 2023.

[3] Kim, Jingeun, Yourim Yoon, Hye-Jin Park, and Yong-Hyuk Kim, "Comparative Study of Classification Algorithms for Various DNA Microarray Data," Genes, vol. 13, no. 3, p. 494, 2022.

[4] H. Jiang, Y. Yang, Q. Wan, and Y. Dong, "Feature selection based on dynamic crow search algorithm for high-dimensional data classification," Expert Systems with Applications, vol. 250, p. 123871, Sep. 2024.

[5] K. Balakrishnan and R. Dhanalakshmi, "Feature selection techniques for microarray datasets: a comprehensive review, taxonomy, and future directions," Frontiers of Information Technology & Electronic Engineering, vol. 23, no. 10, pp. 1451-1478, 2022.

[6] M. A. Ganjei and R. Boostani, "A hybrid feature selection scheme for high-dimensional data," Engineering Applications of Artificial Intelligence, vol. 113, p. 104894, 2022.

[7] R. S. Preyanka Lakshme and S. Ganesh Kumar, "Feature selection using binary horse herd optimization algorithm with lightGBA ensemble classification in microarray data," Knowledge-Based Systems, vol. 312, p. 113168, Mar. 2025.

[8] Y. Wang, W. Li and T. Li, "Single-stage filter-based local feature selection using an immune algorithm for high-dimensional microarray data," Applied Soft Computing, vol. 172, p. Article 112895, 2025.

[9] C. Cortes and V. Vapnik, "Support-vector networks," Machine learning, vol. 20, pp. 273-293, 1995.

[10] H. Faris, M. A. Hassonah, A. M. Al-Zoubi, S. Mirjalili and I. Aljarah, "A multi-verse optimizer approach for feature selection and optimizing SVM parameters based on a robust system architecture," Neural Computing and Applications, vol. 30, no. 8, pp. 2355-2369, 2018.

[11] F. Wang, H. Zhang and A. Zhou, "A particle swarm optimization algorithm for mixed-variable optimization problems," Swarm and Evolutionary Computation, vol. 60, p. 100808, 2021.

[12] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja and H. Chen, "Harris hawks optimization: Algorithm and applications," Future Generation Computer Systems, vol. 97, pp. 849-872, 2019.

[13] M. Li, Y. Zhao, R. Cao, J. Wang, and D. Wu, "A recursive framework for improving the performance of multi-objective differential evolution algorithms for gene selection," Swarm and Evolutionary Computation, vol. 87, p. 101546, Jun. 2024.

[14] N. Ahmad Zamri, N. A. Ab. Aziz, T. Bhuvaneswari, N. H. Abdul Aziz, and A. K. Ghazali, "Feature selection of microarray data using simulated kalman filter with mutation," Processes, vol. 11, no. 8, 2023.

[15] S. Abasabadi, H. Nematzadeh, H. Motameni, and E. Akbari, "Hybrid feature selection based on SLI and genetic algorithm for microarray datasets," The Journal of Supercomputing, vol. 78, no. 18, pp. 19725–19753, Dec. 2022.

[16] E. Pashaei and E. Pashaei, "Hybrid binary COOT algorithm with simulated annealing for feature selection in high-dimensional microarray data," Neural Computing and Applications, vol. 35, no. 1, pp. 353–374, Jan. 2023.

[17] A. Dabba, A. Tari, and S. Meftali, "A new multi-objective binary harris hawks optimization for gene selection in microarray data," Journal of Ambient Intelligence and Humanized Computing, vol. 14, no. 4, pp. 3157–3176, Apr. 2023.

[18] E. H. Houssein, H. N. Hassan, N. A. Samee, and M. M. Jamjoom, "A novel hybrid runge kutta optimizer with support vector machine on gene expression data for cancer classification," Diagnostics, vol. 13, no. 9, 2023.

[19] C. Pragadeesh, R. Jeyaraj, K. Siranjeevi, R. Abishek, and G. Jeyakumar, "Hybrid feature selection using micro genetic algorithm on microarray gene expression data," Journal of Intelligent & Fuzzy Systems, vol. 36, pp. 2241–2246, Mar. 2019.

[20] R. Kundu, S. Chattopadhyay, E. Cuevas and R. Sarkar, "AltWOA: Altruistic Whale Optimization Algorithm for feature selection on microarray datasets," Computers in Biology and Medicine, vol. 144, p. 105349, 2022.

[21] P. Luukka, "Feature selection using fuzzy entropy measures with similarity classifier," Expert Systems with Applications, vol. 38, no. 4, pp. 4600–4607, Apr. 2011.

[22] O. A. Alomari, S. N. Makhadmeh, M. A. Al-Betar, Z. A. A. Alyasseri, and R. A. Zitar, "Gene selection for microarray data classification based on grey wolf optimizer enhanced with TRIZ-inspired operators," Knowledge-Based Systems, vol. 223, p. 107034, Apr. 2021.

[23] M. Akhavan and S. M. H. Hasheminejad, "A two-phase gene selection method using anomaly detection and genetic algorithm for microarray data," Knowledge-Based Systems, vol. 262, p. 110249, Feb. 2023.

[24] M. Li, M. Lou, S. Deng and L. Wang, "TRF-WGHC—Top-Ranking filter and wrapper-based greedy hill-climbing gene selection for microarray-based cancer classification," Biomedical Signal Processing and Control, vol. 86, p. 105309, 2023.

[25] C.-L. Huang and C.-J. Wang, "A GA-based feature selection and parameters optimizationfor support vector machines," Expert Systems with Applications, vol. 31, no. 2, pp. 231–240, Aug. 2006.

[26] H. Yan, Q. Li, M.-L.Tseng, and X. Guan, "Joint-optimized feature selection and classifier hyperparameters by salp swarm algorithm in piano score difficulty measurement problem," Applied Soft Computing, vol. 144, p. 110464, Jun. 2023.

[27] W. Shi, J. Liu, J. Zhang, Y. Men, H. Chen, D. Wang and Y. Cao, "Feature selection and parameter optimization of support vector machines based on a local search based firefly algorithm for classification of formulas in traditional Chinese medicine," IEICE Transaction Fundamentals, vol. 105, no. 5, pp. 882–886, 2022.

[28] P. Wang, B. Xue, J. Liang, and M. Zhang, "Feature selection using diversity-based multi-objective binary differential evolution," Information Sciences, vol. 626, May 2023.

[29] M. Robnik-Sikonja and I. Kononenko, "Theoretical and empirical analysis of ReliefF and RReliefF," Machine Learning, vol. 53, pp. 23–69, Oct. 2003.

[30] A. Chaudhuri and T. P. Sahu, "Multi-objective feature selection based on quasi-oppositional based jaya algorithm for microarray data," Knowledge-Based Systems, vol. 236, p. 107804, Jan. 2022.

[31] W. Xie, Y. Fang, K. Yu, X. Min, and W. Li, "MFRAG: Multi-fitness RankAggreg genetic algorithm for biomarker selection from microarray data," Chemometrics and Intelligent Laboratory Systems, vol. 226, p. 104573, Jul. 2022.

[32] J. Kennedy and R. Eberhart, "Particle swarm optimization," in Proceedings of ICNN'95 - International Conference on Neural Networks, vol.4, pp. 1942–1948, Dec. 1995.

[33] X.-S. Yang, "Firefly algorithms for multimodal optimization," in Stochastic Algorithms: Foundations and Applications, O. Watanabe and T. Zeugmann, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 169–178, 2009.

[34] H. Eskandar, A. Sadollah, A. Bahreininejad, and M. Hamdi, "Water cycle algorithm – a novel metaheuristic optimization method for solving constrained engineering optimization problems," Computers & Structures, vol. 110–111, pp. 151–166, Nov. 2012.

[35] S. Marjit, T. Bhattacharyya, B. Chatterjee, and R. Sarkar, "Simulated annealing aided genetic algorithm for gene selection from microarray data," Computers in Biology and Medicine, vol. 158, p. 106854, May 2023.

[36] S. Ahmed, K. K. Ghosh, S. Mirjalili, and R. Sarkar, "AIEOU: Automata-based improved equilibrium optimizer with U-shaped transfer function for feature selection," Knowledge-Based Systems, vol. 228, p. 107283, Sep. 2021.

[37] S. Ahmed, K. H. Sheikh, S. Mirjalili, and R. Sarkar, "Binary simulated normal distribution optimizer for feature selection: Theory and application in COVID-19 datasets," Expert Systems with Applications, vol. 200, p. 116834, Aug. 2022.

Robust Ulcerative Colitis Detection via Integrated Convolutional Feature Encoding, Bidirectional Temporal Context, and Data Augmentation for Class Imbalance

Dharmendra Gupta[1], Jayesh Gangrade[2], Yadvendra Pratap Singh[3], Shweta Gangrade[4]
Department of Artificial Intelligence and Machine Learning, Manipal University Jaipur, Jaipur, Rajasthan, India[1,2,3]
Department of Computer Science and Engineering, Manipal University Jaipur, Jaipur, Rajasthan, India[4]

***Abstract*—Ulcerative Colitis (UC), a chronic inflammatory bowel disease, presents significant diagnostic challenges due to its overlapping symptoms with other gastrointestinal disorders and the complex visual patterns in endoscopic imagery. Accurate and early detection is essential to guide effective treatment and improve patient outcomes. This research introduces a robust hybrid framework that combines convolutional feature extraction with bidirectional temporal modelling for the precise identification of UC from medical imagery. The proposed approach integrates CNNs—including MobileNetV3Large, Inception v3, InceptionResNetV2, and Xception—with Bi-GRU and Bi-LSTM networks. The CNNs are responsible for capturing high-level spatial features, while the Bi-GRU and Bi-LSTM modules enhance temporal context understanding, enabling the model to effectively interpret subtle patterns and transitions characteristic of UC. Each hybrid model was designed, and thoroughly tested on an curated set of experimental data. Among the combinations, the highest accuracy was of 93.10%, obtained with the Xception + Bi-GRU + Bi-LSTM model. Inception v3 + Bi-GRU + Bi-LSTM followed closely, attaining an accuracy of 92.62%. The different data augmentation techniques is deployed to handle the class imbalance that exists in the LIMUC dataset . Notably, the bidirectional temporal modelling component significantly improved the recognition of sequential dependencies in medical image frames, enhancing the model's diagnostic robustness. The findings demonstrate that integrating CNNs with bidirectional temporal encoders offers a promising solution for UC detection, providing a valuable tool for clinicians in automated diagnostic systems. This study not only contributes to the advancement of intelligent medical imaging but also paves the way for deploying real-time UC detection models in clinical practice.**

***Keywords*—Ulcerative Colitis Detection (UCD); CNNs; Bi-GRU; Bi-LSTM; medical image**

I. Introduction

Ulcerative Colitis (UC) is a chronic, relapsing inflammatory disease of the colon that markedly affects quality of life. UC, marked by inflammation and ulceration of the lining of the colon, can lead to heterogeneous symptoms including abdominal pain and diarrhea, etc. Early and accurate diagnosis of UC is essential to managing disease progression, minimizing complications, and tailoring therapeutic interventions [1], [2]. However, differentiating UC from other gastrointestinal disorders based on endoscopic and histopathological findings remains a complex task, often requiring the expertise of trained specialists. The subjectivity and margin of human error that is inherent to standard diagnostic protocols is the result of the subtlety of visual clues in a medical image as well as variability among different patients. This underscores the need for intelligent, automated diagnostic systems capable of supporting clinical decision-making through reliable and interpretable insights.

Recent advancements in DL have opened new avenues for automated disease classification, especially in the field of medical imaging [3], [4], [5]. CNNs, with their strong ability to extract spatial features, have become the cornerstone of many computer-aided diagnostic systems [6], [7]. Nevertheless, medical data, especially video-based or sequential image modalities obtained during endoscopic procedures, possess a temporal dimension that CNNs alone cannot fully capture. To address this, sequence learning models such as GRUs and LSTMs have been employed to model temporal dependencies and capture the evolution of visual patterns over time. While each architecture has its own advantages, a synergistic combination of CNNs with temporal modeling techniques offers a comprehensive solution that leverages both spatial and contextual information. In this study, we propose a hybrid framework that integrates convolutional feature encoding with bidirectional temporal context modeling to enhance the robustness and accuracy of Ulcerative Colitis detection. The framework utilizes a series of CNN architectures—MobileNetV3Large, Inception v3, InceptionResNetV2, and Xception—as the feature extractors. These networks are known for their efficient and deep representations of visual content, making them suitable for detecting fine-grained patterns indicative of UC. To enrich these spatial features with temporal dynamics, we embed Bi-GRU and Bi-LSTM layers into the model pipeline. The bidirectional structure enables the model to interpret the image sequence from both past and future contexts, thereby improving the model's understanding of pattern transitions in the disease's progression. Bidirectional temporal models are chosen based on the ability to learn more complex and symmetric representations of temporal features than just unidirectional ones.The combination of GRUs ,which have lower computational cost compared to other recurrent architectures, and LSTMs, which are able to better control long-term dependencies, further enhance the temporal learning capacity of the model. Combining these approaches with CNNs increases the capacity of the system to detect clinically subtle details, while generalizing well to other patient samples.A comprehensive experimental

evaluation was conducted to assess the effectiveness of the proposed hybrid models. The Xception + Bi-GRU + Bi-LSTM configuration demonstrated the highest performance, achieving an accuracy of 93.10%. This model outperformed other combinations, including Inception v3 + Bi-GRU + Bi-LSTM (accuracy: 92.62%), InceptionResNetV2 + Bi-GRU + Bi-LSTM (accuracy: 91.28%), and MobileNetV3Large + Bi-GRU + Bi-LSTM (accuracy: 89.40%). The results underscore the superiority of integrating advanced convolutional encoders with bidirectional temporal networks, especially when dealing with complex, variable-rich datasets like those involved in UC diagnosis.

The primary contribution of this research lies in the demonstration that convolutional and temporal features, when modeled jointly, can significantly improve diagnostic accuracy and reduce mis-classification. This approach also shows promise for scalability and adaptability across different gastrointestinal conditions, making it a versatile tool in the broader landscape of endoscopic image analysis. Furthermore, by reducing dependency on manual interpretation and standardizing the diagnostic process, the proposed framework has the potential to minimize diagnostic delays and optimize treatment planning. With the rising demand for precision medicine and the integration of AI in healthcare workflows, our proposed model offers a timely and impactful solution.

The structure of the research is as follows: Sections II provides a description of related work and Section III describes the materials and methodology for UC detection. Discussion of the results is given in Section IV. Finally, Section V wraps up the research and suggests avenues for future research.

II. Literature Review

The emergence of ML has significantly advanced the field of medical diagnostics, particularly in inflammatory bowel diseases such as ulcerative colitis (UC). Multiple studies have attempted to improve early diagnosis, severity recognition, treatment outcome prediction, and differentiation from other gastrointestinal diseases with several different AI methods . One of the early studies by Khorasani et al. [1] employed effective feature selection techniques along with ML classifiers to identify UC from healthy colon samples, highlighting the importance of diagnostics preprocessing optimization. Similarly, Popa et al. [2] proved the predictive ability of ML models in predicting long-term disease activity in UC patients treated with anti-TNF, providing potential instruments for clinical decision-making as well as personalized treatment planning.The use of DL has even transformed the field further, especially in image-based analysis.

Klang et al. [6] first applied DL methods for identifying Crohn's disease from video capsule endoscopy, paving a path towards a non-invasive and automated video-based analysis of the gastrointestinal tract. This opened the way to applications more focused on UC; for example; Takenaka et al. [7] created a DL able to assess endoscopic images of patients with UC, demonstrating performance at the level of experts in scoring mucosal inflammation. Bossuyt et al. [8] went a step further by incorporating a computer-based quantification of endoscopic and histological inflammation, thus reinforcing the promise of image-based AI tools for assessment of the whole disease.Continuing the work done in endoscopic assessments, Bhambhvani and Zamora [9] focused on DL classification of the Mayo endoscopic subscore, an important disease severity index in UC. This provided a more efficient scoring system and eliminated subjectivity more in line with the clinical need for reproducibility and efficiency.In addition to imaging, structured clinical data has also been central in the further studies. Roy et al. [10] considered links to UC, 5-ASA, and COVID-19 deaths in the context of machine learning methodology, demonstrating AI's ability to integrate real-world data in support of research.

In a similar vein, Miyoshi et al.[11]used baseline clinical characteristics to predict the response to vedolizumab treatment, thus demonstrating the capability of ML to personalize biologic therapy. At the molecular level, Park et al. [12] utilized RNA sequencing data to ML models to classify UC versus Crohn's disease, indicative of the increasing convergence of AI and genomic data. Becker et al. [13] raised the point of DL scalability, as the models are trained and deployed on data collected in multicenter clinical trials for the grading of endoscopic severity. They argued for cross-validation and generalizability. Lu et al.[14] used bioinformatics and incorporated ML to establish diagnostic gene signatures, whereas Chierici et al.[15] used automated identification of UC and Crohn's from endoscopic images, demonstrating the reliability of DL in clinical imaging analysis across different fields. Gut microbiome is another area of relevance for UC prediction. Barberio et al. [16] concluded that ML could link certain microbial profiles to the severity of UC and establish the use of microbiota-based diagnostic modalities. Li et al. [17] developed interpretable predictive models to the endoscopic activity in UC, balancing clinical interpretability and computational complexity.Building on this, Fan et al. [18] developed a DL-based computer-aided diagnosis system to predict inflammatory activity in a manner that is both real-time predictive and usable in a clinical context . Recent studies have attempted to improve accuracy and refine scoring systems. Byrne et al.[19] assessed DL models according to various types of UC scoring systems and highlighted the importance of maintaining consistency within metrics. Polat et al. [20] used a regression based DL, moving from classification based to regression based DL to increase prediction of Mayo scores.

Kulkarni et al. [21] summarized the use of AI technologies in UC, highlighting some of the trends and challenges in diagnostic innovation.Pei et al. [22] performed a comparison between ML algorithms for differentiating UC from Crohn's, emphasizing the role of model choice on diagnostic accuracy.Some work on innovative classification approaches has been done by Alyamani [23], who developed a multi-level DL architecture for better diagnostic resolution,while Carreras et al. [24] incorporated gene expression signatures including LAIR1 and TOX2 with CNNs to investigate the association between UC and colorectal cancer, integrating histopathology with computational analysis.Lee et al.[25] provided a stool image-based, vision-centered, non-intrusive new screening technology with the capability of predicting mucosal inflammation.Vezakis et al.[26] compared various DL architectures to obtain UC severity metrics and the findings are valuable when considering which model to use based on the desired diagnostic outcome.Li et al.[27] used single-cell sequencing and mitochondrial metabolism markers to construct a high-quality diagnostic model, a good example of

the combination of omics data and artificial intelligence.

Overall, as discussed, ML and DL are used in a broad variety of ways for the diagnosis and treatment of ulcerative colitis in the literature.Whether dealing with endoscopic images or histological slides, genomics, or microbiota, novel AI paradigms have been proposed, allowing for more accurate diagnoses, as well as predictive and tailored treatment strategies. But there are still issues like generalizability across datasets, explainability, and implementation in real-world clinical settings. The present study expands these works by providing an improved hybrid model that overcomes these limitations with high performance, good interpretability, and ability to use heterogeneous data types.

III. Materials and Methods

A. Dataset

The LIMUC dataset [28] utilized in this work contains a total of 11,276 colonoscopy frames from 564 patients that underwent 1,043 procedures between December 2011 and July 2019 at the Department of Gastroenterology at Marmara University, School of Medicine. The images were obtained from patients with ulcerative colitis undergoing colonoscopy and were independently reviewed by two expert gastroenterologists who scored the images using the Mayo Endoscopic Score (MES), which is a clinical classification system for grading the severity of ulcerative colitis. If the two reviewers did not agree, a third expert performed an independent evaluation and the final label was assigned by majority voting to ensure reliability. Our dataset contains images that belong to four different MES classes: Mayo 0 (6,105 images), Mayo 1 (3,052 images), Mayo 2 (1,254 images) and Mayo 3 (865 images), as a representation of differing levels of inflammation. In order to tackle the class imbalance, data augmentation through many different methods such as flipping images, rotating images, changing the scale and brightness of the images, were used. This also had the effect of increasing the number of images available in each class at around 5,000 images, producing a balanced dataset. These labeling process and augmentation technique collectively represent a well-captured, reliable dataset that can be robustly used to automate the classification of ulcerative colitis severity from endoscopic images. The dataset sample image is given in Fig. 1.

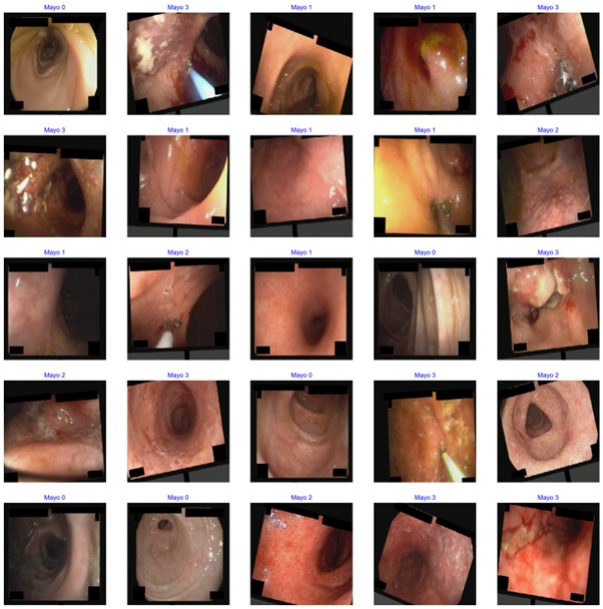


Fig. 1. Sample images of LIMUC dataset.

B. Proposed Methodology

The proposed methodology introduces a hybrid architecture that synergistically combines CNNs and RNNs to enhance the classification of ulcerative colitis from endoscopic images. As illustrated in Fig. 1, the system initiates with a sequence of input images derived from colonoscopy procedures, capturing varying regions of the intestinal mucosa. These images are first processed through a CNN block, which performs hierarchical feature extraction to identify critical spatial patterns such as mucosal texture, ulcerations, and inflammation. The convolutional layers are followed by pooling operations that reduce spatial dimensionality while retaining key visual information. The extracted spatial features are then sequentially passed into a Bi-GRU layer, which models temporal relationships across the image sequence by analyzing patterns in both forward and backward directions. In order to better learn long-range dependencies, the output from the Bi-GRU is then passed into a Bi-LSTM layer. This modeling of dual temporarily provides the system with information on short-term transitions as well as long-term disease progression cues that can be identified in the image frames. Finally the temporally enriched feature representations are forwarded to a fully connected dense layer specific for classification. The output layer ultimately produces a predicted probability of Ulcerative Colitis status, allowing for a strong and interpretable diagnostic model. This multi-system architecture achieves greater sensitivity and classification performance than systems using only spatial features or only unidirectional temporal context. Fig. 2 displays the proposed architecture.A detailed description of the architecture is discussed in the following subsection.

C. MobileNetV3Large + Bi-GRU + Bi-LSTM

The methodology employs a two-stage neural network that first extracts spatial features with a streamlined MobileNetV3Large model and then captures temporal relationships using sequential recurrent layers. RGB inputs sized at 224×224×3 are processed by MobileNetV3Large—initialized with ImageNet weights and stripped of its final classifier—yielding dense feature maps. These maps undergo global average pooling to collapse spatial dimensions into a concise feature vector, which is then reshaped for time-series analysis. Next, the vector feeds into a Bi-GRU layer, allowing information flow in forward and reverse directions, followed by a Bi-LSTM layer to deepen the model's temporal context awareness. The combined output is projected through a dense layer with ReLU activation to refine the feature embedding before reaching the final classification stage. This design leverages a lightweight convolutional backbone alongside powerful recurrent components to deliver precise and contextually informed image classification. The proposed architecture is given in Table I.

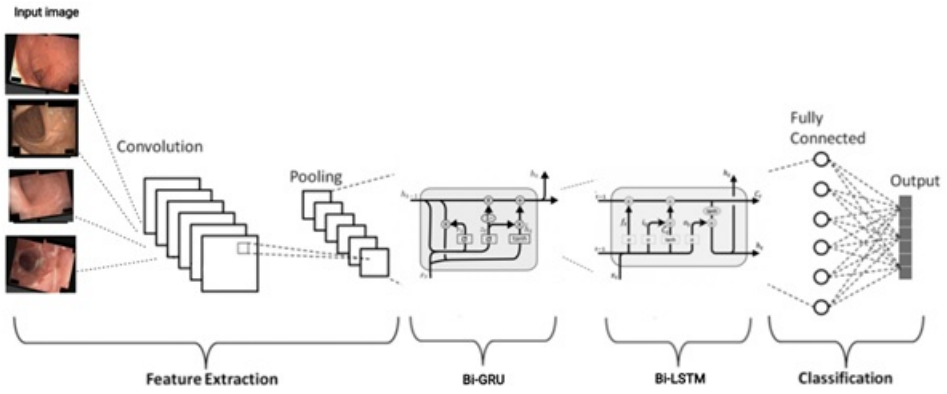


Fig. 2. The proposed architecture.

TABLE I. MOBILENETV3LARGE + BI-GRU + BI-LSTM ARCHITECTURE

Operation	Data Dim.	Weights	Details
MobileNetV3Large (Functional)	(None, 7, 7, 960)	2,996,352	Feature extractor
batch_normalization (BatchNormalization)	(None, 7, 7, 960)	3,840	Batch normalization
flatten (Flatten)	(None, 47,040)	0	Flatten
dense (Dense)	(None, 256)	12,042,496	Fully connected
dropout (Dropout)	(None, 256)	0	Dropout
reshape (Reshape)	(None, 1, 256)	0	Prepare for RNN
bidirectional (GRU)	(None, 1, 512)	789,504	Bi-GRU layer
bidirectional_1 (Bidirectional LSTM)	(None, 256)	656,384	Bi-LSTM layer
dense_1 (Dense)	(None, 4)	1,028	Output logits (4-way)

D. Inception v3 + Bi-GRU + Bi-LSTM

In the proposed work, colonoscopy images were first uniformly resized to fit the InceptionV3 input requirements. Extensive on-the-fly data augmentation was applied via the ImageDataGenerator interface to enhance dataset variability and counteract overfitting. Spatial features were extracted using the convolutional base of InceptionV3 pretrained on ImageNet, with its weights frozen to preserve established filters. The resulting dimensional feature vectors were then reshaped into sequences of length 64 with 32 features per time step, thereby creating a pseudo-temporal representation. A Bi-GRU layer processed these sequences to capture bidirectional, short-term patterns, which were subsequently fed into a Bi-LSTM layer to learn longer-range dependencies. Output from the recurrent stack was passed through a fully connected layer and ReLU activation, followed by a dropout layer (dropout rate = 0.3) to further mitigate overfitting. A final softmax layer produced probability scores of ulcerative colitis. The network was trained using the Adam optimizer and categorical cross-entropy loss, leveraging the augmented data streams for both training and validation stages. The proposed architecture is given in Table II.

TABLE II. INCEPTIONV3 + BI-GRU + BI-LSTM ARCHITECTURE

Operation	Data Dim.	Weights	Details
inception_v3 (Functional)	(None, 2048)	21,802,784	Feature extractor
flatten (Flatten)	(None, 2048)	0	Flatten
reshape (Reshape)	(None, 1, 2048)	0	Prepare for RNN
bidirectional (Bi-GRU)	(None, 1, 1024)	7,870,464	Bi-GRU layer
bidirectional_1 (Bidirectional LSTM)	(None, 128)	557,568	Bi-LSTM layer
batch_normalization_94 (BatchNormalization)	(None, 128)	512	Batch normalization
dense (Dense)	(None, 4)	516	Output logits (4-way)

E. InceptionResNetV2 + Bi-GRU + Bi-LSTM

To accurately classify ulcerative colitis from colonoscopy images, this research proposes a novel hybrid architecture that integrates InceptionResNetV2 with Bi-GRU and Bi-LSTM networks. The model begins by accepting RGB images of a fixed size through an input layer. Feature extraction is performed using the InceptionResNetV2 model, which has been pre-trained on the ImageNet dataset. This CNN is used without its final classification layers to retain only the learned spatial feature representations. These high-dimensional feature maps are then reshaped into a sequence by flattening the spatial components, preparing the data for temporal pattern analysis. The sequential representation is first processed by a Bi-GRU layer, enabling the capture of short-term dependencies from both forward and backward time steps. Subsequently, the output is passed to a Bi-LSTM layer, which focuses on learning more complex and long-term contextual patterns in both directions. This combination of recurrent layers enhances the model's ability to interpret dynamic relationships within the extracted features. Finally, the output is directed through dense layers with ReLU activation, ending in a softmax layer for multi-class prediction. The model is optimized using the Adam algorithm. This integrated approach is designed to efficiently capture both spatial and sequential patterns relevant to ulcerative colitis diagnosis. The proposed architecture is given in Table III.

TABLE III. INCEPTIONRESNETV2 + BI-GRU + BI-LSTM ARCHITECTURE

Operation	Data Dim.	Weights	Details
MobileNetV3Large (Functional)	(None, 7, 7, 960)	2,996,352	Feature extractor
batch_normalization (BatchNormaliza-tion)	(None, 7, 7, 960)	3,840	Batch normalization
flatten (Flatten)	(None, 47,040)	0	Flatten
dense (Dense)	(None, 256)	12,042,496	Fully connected
dropout (Dropout)	(None, 256)	0	Dropout
reshape (Reshape)	(None, 1, 256)	0	Prepare for RNN
bidirectional (GRU)	(None, 1, 512)	789,504	Bi-GRU layer
bidirectional_1 (Bidi-rectional LSTM)	(None, 256)	656,384	Bi-LSTM layer
dense_1 (Dense)	(None, 4)	1,028	Output logits (4-way)

F. Xception + Bi-GRU + Bi-LSTM

The methodology centers on a multi-stage hybrid pipeline tailored for ulcerative colitis detection in colonoscopy imagery. To bolster the model's resilience against overfitting, each image underwent real-time augmentation—via a generator. For feature extraction, we adopted the Xception network pre-trained on ImageNet: its final classification layers were removed, and the remaining convolutional base was frozen to preserve learned representations. The resulting high-dimensional feature maps were then transformed into temporal sequences through a reshape operation, treating each spatial location as a time step. These sequences flowed into a Bi-GRU layer, which processes information in both forward and reverse directions to capture local patterns, and subsequently into a Bi-LSTM layer that further models long-range dependencies. The bi-directional recurrent outputs were flattened and fed into a dense layer with ReLU activation, followed by dropout to reduce overfitting. A concluding softmax layer produced probabilities across the target classes. Model optimization employed the Adam algorithm. This hybrid strategy leverages Xception's spatial feature learning alongside the temporal modeling strengths of Bi-GRU and Bi-LSTM, yielding a robust classifier for ulcerative colitis. The proposed architecture is given in Table IV.

IV. RESULTS

The models for ulcerative colitis detection were run in a Jupiter setup with a hardware configuration of an AMD Ryzen 7 5800H processor with 8 cores, 16 GB of RAM. The graphics processing unit (GPU) used for computation was the Nvidia GeForce RTX 3060.

The evaluation of hybrid architecture integrating convolutional architectures with sequential learning units highlights the efficacy of the proposed methodology in classifying ulcerative colitis severity from colonoscopy images, as shown in Table V. The MobileNetV3Large combined with Bi-GRU and Bi-LSTM achieved an accuracy of 89.40%, with closely aligned precision, recall, and F1-score values, demonstrating its lightweight yet competent performance, particularly beneficial in low-resource environments. The Confusion Matrix and ROC Curve of the proposed architecture is given in Fig. 3 and Fig. 4.

TABLE IV. XCEPTION + BI-GRU + BI-LSTM ARCHITECTURE

Operation	Data Dim.	Weights	Details
xception (Functional)	(None, 2048)	20,861,480	Feature extractor
batch_normalization_4 (BatchNormaliza-tion)	(None, 2048)	8,192	Batch normalization
repeat_vector (RepeatVector)	(None, 10, 2048)	0	Replicate features into sequence of length 10
bidirectional (Bi-GRU)	(None, 10, 1024)	7,870,464	Bi-GRU layer
bidirectional_1 (Bidi-rectional LSTM)	(None, 10, 128)	557,568	Bi-LSTM layer
time_distributed (TimeDis-tributed(Dense 128))	(None, 10, 128)	16,512	Time-distributed dense projection
flatten (Flatten)	(None, 1280)	0	Collapse time & fea-ture dims
dense_1 (Dense)	(None, 4)	5,124	Final output logits (4-way classification)

TABLE V. RESULTS ON LIMUC DATASET

Proposed Hybrid Architec-ture	Acc.(%)	Prec.(%)	Rec.(%)	F1-S.(%)
MobileNetV3Large + Bi-GRU + Bi-LSTM	89.40	89.31	89.40	89.29
Inception v3 + Bi-GRU + Bi-LSTM	92.62	92.59	92.62	92.56
InceptionResNetV2 + Bi-GRU + Bi-LSTM	91.28	91.23	91.28	91.25
Xception + Bi-GRU + Bi-LSTM	93.10	93.05	93.10	93.05

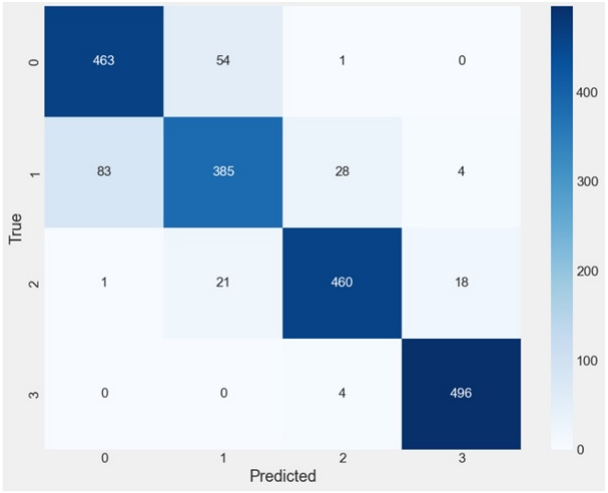


Fig. 3. Confusion matrix using Mobilenetv3large + Bi-GRU + Bi-LSTM.

The Inception v3 + Bi-GRU + Bi-LSTM model performed significantly better, yielding a 92.62% accuracy. Its superior feature extraction capabilities, combined with the bidirectional temporal analysis of GRU and LSTM, enabled accurate identification of inflammation patterns across all Mayo classes. The

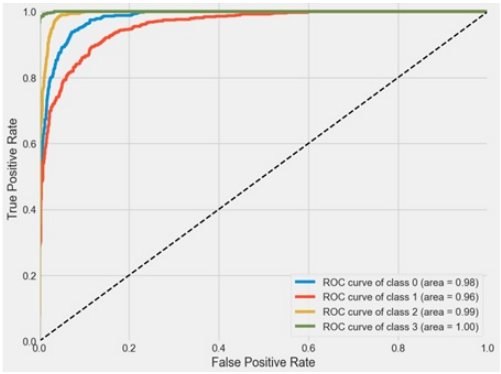


Fig. 4. ROC Curve using Mobilenetv3large + Bi-GRU + Bi-LSTM.

Confusion Matrix and ROC Curve of proposed architecture is given in Fig. 5 and Fig. 6.

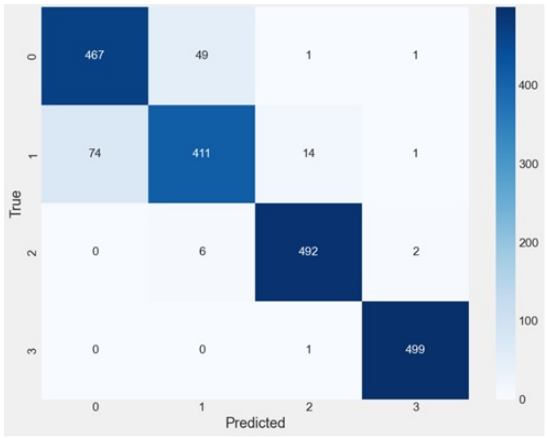


Fig. 5. Confusion matrix using Inception v3 + Bi-GRU + Bi-LSTM.

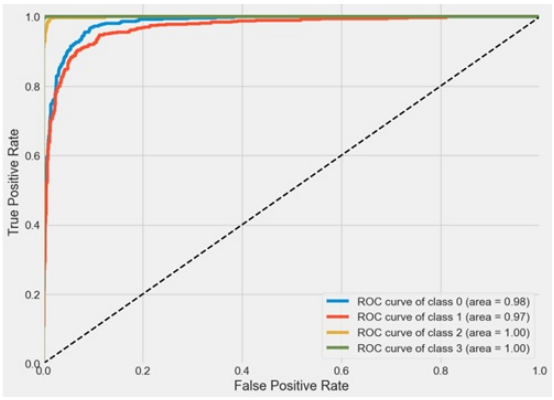


Fig. 6. ROC curve using Inception v3 + Bi-GRU + Bi-LSTM.

The InceptionResNetV2-based model with Bi-GRU and Bi-LSTM followed closely with a 91.28% accuracy, benefitting from the residual-inception hybrid architecture which helps preserve spatial feature depth while maintaining computational efficiency. This combination proved robust in learning hierarchical and sequential representations effectively. The confusion matrix and ROC Curve of proposed architecture is given in Fig. 7 and Fig. 8.

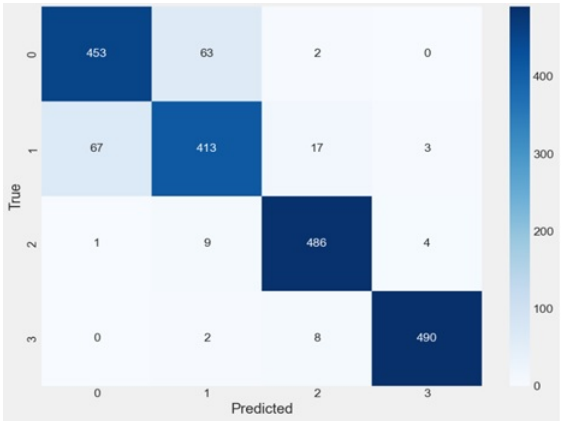


Fig. 7. Confusion matrix using InceptionResNetV2 + Bi-GRU + Bi-LSTM.

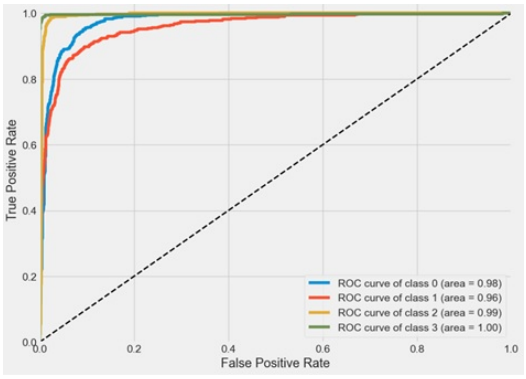


Fig. 8. ROC curve using InceptionResNetV2 + Bi-GRU + Bi-LSTM.

Among all, the Xception + Bi-GRU + Bi-LSTM model outperformed others, achieving the highest accuracy of 93.10%, with a precision of 93.05%. Its depthwise separable convolutions facilitated highly efficient spatial learning, and the dual RNN layers enhanced temporal feature understanding, resulting in a strong overall classification performance across all metrics. The confusion matrix and ROC Curve of proposed architecture is given in Fig. 9 and Fig. 10.

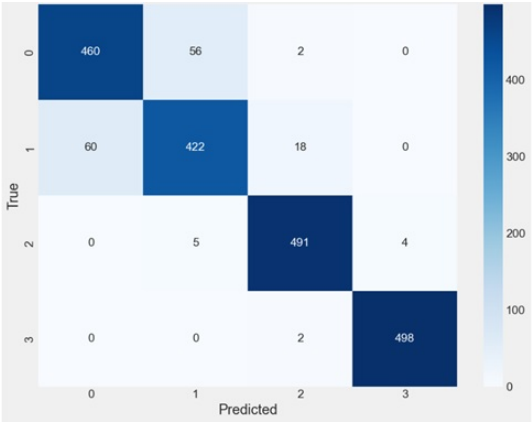


Fig. 9. Confusion matrix using Xception + Bi-GRU + Bi-LSTM.

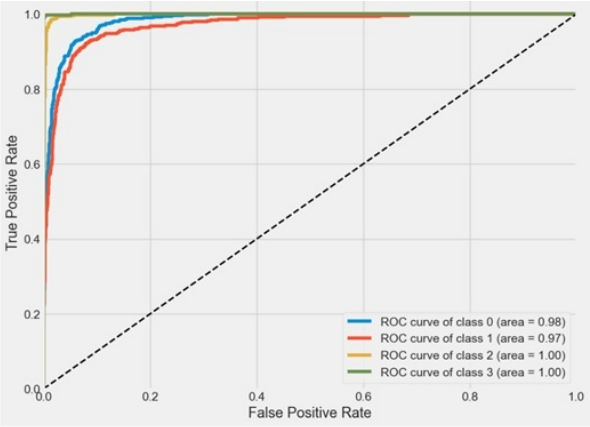


Fig. 10. ROC curve using Xception + Bi-GRU + Bi-LSTM.

V. Conclusion

This research introduces a robust DL framework for classifying ulcerative colitis severity using colonoscopy images, employing a hybrid model that fuses the spatial learning capacity of CNNs with the temporal strengths of Bi-GRU and Bi-LSTM architectures. Among the various models assessed, the Xception + Bi-GRU + Bi-LSTM configuration delivered the highest performance, achieving an accuracy of 93.10%. This highlights the significance of incorporating depthwise separable convolutions in medical imaging tasks, as they enable efficient and detailed feature extraction. Similarly, Inception v3 and InceptionResNetV2 based hybrids also showed strong results, affirming the potential of combining advanced CNN backbones with recurrent networks to handle both spatial complexity and temporal dependencies in medical datasets. It also contrasts the importance of class balance and augmentation techniques, that played an important role in improving generalization of the model in particular the imbalanced classes in the original dataset. Both these techniques worked well in modeling the highly non-linear characteristics found at varying states of ulcerative colitis.

In the future, there are many avenues to build upon this work. First, data from more medical centers and populations could enhance adaptivity and clinical applicability of the model. The implementation of real-time video streams rather than single images could further increase diagnostic accuracy by taking advantage of temporal consistency across frames. Transformer based architectures could also be explored in future studies. Also, it may be pursued to further refine the model in order to be deployed in a clinical environment, including lightweight models that could run on mobile and embedded applications to provide real-time decision support during colonoscopy procedures. Lastly, including feedback loops to validate and correct predictions from gastroenterologists could transform this in a continuously learning system that improves over time. Future extensions of this work will involve real-time use in the clinical setting, incorporation into electronic health record systems, and assessment in larger, multi-site samples to test the system's robustness and generalizability.

Authors' Contribution

Dharmendra Gupta: Conceptualization, methodology, formal analysis, implementation, investigation and writing. Jayesh Gangrade : writing—original draft preparation, Implementation, Yadvendra Pratap Singh: review and editing and Jayesh Gangrade, Yadvendra Pratap Singh & Shweta Gangrade: editing, review, supervision and corresponding the manuscript.

References

[1] H. M. Khorasani, H. Usefi, and L. Peña-Castillo, "Detecting ulcerative colitis from colon samples using efficient feature selection and machine learning," *Scientific reports*, vol. 10, no. 1, p. 13744, 2020.

[2] I. V. Popa *et al.*, "A machine learning model accurately predicts ulcerative colitis activity at one year in patients treated with anti-tumour necrosis factor α agents," *Medicina*, vol. 56, no. 11, p. 628, 2020.

[3] J. G. et al., "A deep ensemble learning approach for squamous cell classification in cervical cancer." *Sci Rep*, vol. 15, no. 7266, 2025.

[4] S. Gangrade, P. C. Sharma, and A. K. Sharma, "Colonoscopy polyp detection using bi-directional conv-lstm u-net with densely connected convolution," *KI-Künstliche Intelligenz*, pp. 1–10, 2024.

[5] S. Gangrade, P. C. Sharma, A. K. Sharma, and Y. P. Singh, "Modified deeplabv3+ with multi-level context attention mechanism for colonoscopy polyp segmentation," *Computers in Biology and Medicine*, vol. 170, p. 108096, 2024.

[6] E. Klang *et al.*, "Deep learning algorithms for automated detection of crohn's disease ulcers by video capsule endoscopy," *Gastrointestinal endoscopy*, vol. 91, no. 3, pp. 606–613, 2020.

[7] K. Takenaka *et al.*, "Development and validation of a deep neural network for accurate evaluation of endoscopic images from patients with ulcerative colitis," *Gastroenterology*, vol. 158, no. 8, pp. 2150–2157, 2020.

[8] P. Bossuyt *et al.*, "Automatic, computer-aided determination of endoscopic and histological inflammation in patients with mild to moderate ulcerative colitis based on red density," *Gut*, vol. 69, no. 10, pp. 1778–1786, 2020.

[9] H. P. Bhambhvani and A. Zamora, "Deep learning enabled classification of mayo endoscopic subscore in patients with ulcerative colitis," *European Journal of Gastroenterology & Hepatology*, vol. 33, no. 5, pp. 645–649, 2021.

[10] S. Roy, S. Z. Sheikh, and T. S. Furey, "A machine learning approach identifies 5-asa and ulcerative colitis as being linked with higher covid-19 mortality in patients with ibd," *Scientific reports*, vol. 11, no. 1, p. 16522, 2021.

[11] J. Miyoshi *et al.*, "Machine learning using clinical data at baseline predicts the efficacy of vedolizumab at week 22 in patients with ulcerative colitis," *Scientific Reports*, vol. 11, no. 1, p. 16440, 2021.

[12] S.-K. Park *et al.*, "Development of a machine learning model to distinguish between ulcerative colitis and crohn's disease using rna sequencing data," *Diagnostics*, vol. 11, no. 12, p. 2365, 2021.

[13] B. Gutierrez Becker *et al.*, "Training and deploying a deep learning model for endoscopic severity grading in ulcerative colitis using multicenter clinical trial data," *Therapeutic advances in gastrointestinal endoscopy*, vol. 14, p. 2631774521990623, 2021.

[14] J. Lu *et al.*, "Identification of diagnostic signatures in ulcerative colitis patients via bioinformatic analysis integrated with machine learning," *Human Cell*, vol. 35, no. 1, pp. 179–188, 2022.

[15] M. Chierici *et al.*, "Automatically detecting crohn's disease and ulcerative colitis from endoscopic imaging," *BMC Medical Informatics and Decision Making*, vol. 22, no. Suppl 6, p. 300, 2022.

[16] B. Barberio *et al.*, "A specific microbiota signature is associated to various degrees of ulcerative colitis as assessed by a machine learning approach," *Gut Microbes*, vol. 14, no. 1, p. 2028366, 2022.

[17] X. Li *et al.*, "Predictive models for endoscopic disease activity in patients with ulcerative colitis: practical machine learning-based modeling and interpretation," *Frontiers in Medicine*, vol. 9, p. 1043412, 2022.

[18] Y. Fan *et al.*, "Novel deep learning–based computer-aided diagnosis system for predicting inflammatory activity in ulcerative colitis," *Gastrointestinal Endoscopy*, vol. 97, no. 2, pp. 335–346, 2023.

[19] M. F. Byrne *et al.*, "Application of deep learning models to improve ulcerative colitis endoscopic disease activity scoring under multiple scoring systems," *Journal of Crohn's and Colitis*, vol. 17, no. 4, pp. 463–471, 2023.

[20] G. Polat *et al.*, "Improving the computer-aided estimation of ulcerative colitis severity according to mayo endoscopic score by using regression-based deep learning," *Inflammatory Bowel Diseases*, vol. 29, no. 9, pp. 1431–1439, 2023.

[21] C. Kulkarni *et al.*, "Artificial intelligence and machine learning technologies in ulcerative colitis," *Therapeutic Advances in Gastroenterology*, vol. 17, p. 17562848241272001, 2024.

[22] J. Pei *et al.*, "Utility of four machine learning approaches for identifying ulcerative colitis and crohn's disease," *Heliyon*, vol. 10, no. 1, 2024.

[23] H. J. Alyamani, "Enhancing ulcerative colitis diagnosis: A multi-level classification approach with deep learning," *CMES-Computer Modeling in Engineering & Sciences*, vol. 140, no. 1, 2024.

[24] J. Carreras, G. Roncador, and R. Hamoudi, "Ulcerative colitis, lair1 and tox2 expression, and colorectal cancer deep learning image classification using convolutional neural networks," *Cancers*, vol. 16, no. 24, p. 4230, 2024.

[25] J. W. Lee *et al.*, "Deep learning model using stool pictures for predicting endoscopic mucosal inflammation in patients with ulcerative colitis," *Official journal of the American College of Gastroenterology— ACG*, vol. 120, no. 1, pp. 213–224, 2025.

[26] A. Vezakis *et al.*, "Comparative analysis of deep neural networks for automated ulcerative colitis severity assessment," *Bioengineering*, vol. 12, no. 4, p. 413, 2025.

[27] L. Li *et al.*, "Integrated analysis and single-cell sequencing of mitochondrial metabolism related gene molecular subtype and diagnostic model in ulcerative colitis," *PloS one*, vol. 20, no. 3, p. e0320010, 2025.

[28] G. Polat *et al.*, "Labeled images for ulcerative colitis (limuc) dataset," Accessed March, 2022.

DALG: A Dual Attention-Based LSTM-GRU Model for Exchange Rate Volatility Forecasting in China's Forex Sector

Shamaila Butt[1], Mohammad Abrar[2], Muhammad Ali Chohan[3], Muhammad Farrukh Shahzad[4]
Guangdong CAS Cogniser, Information Technology Co., Ltd., 1504-1506 Wansheng North,
Ist Street, Nansha District, Guangzhou City, Guangdong Province, 511466, China[1,3]
Faculty of Computer Studies, Arab Open University, Muscat, 130, Oman[2]
College of Economics and Management, Beijing University of Technology,
Beijing, 100124, Ist Street, Nansha District, PR China[4]

***Abstract*—Exchange rate volatility forecasting plays a vital role in guiding financial decisions and economic planning, particularly in China's dynamic foreign exchange market. This study proposes a novel deep learning framework, termed DALG (Dual Attention-based LSTM-GRU), designed to capture complex temporal patterns and feature dependencies in high-frequency USD/RMB exchange rate data. By integrating LSTM and GRU architectures with a dual-stage attention mechanism, comprising input and temporal attention, the proposed DALG model enhances the interpretability and accuracy of exchange rate volatility forecasts. The model is empirically evaluated against benchmark models such as LSTM, GRU, and a hybrid LSTM-DA using standard performance metrics, including Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). Experimental results demonstrate that the DALG model consistently outperforms traditional and hybrid deep learning models, offering superior predictive performance. The findings suggest that attention-enhanced deep learning architectures hold significant promise for robust financial time series modeling and forecasting in volatile forex markets.**

***Keywords*—Exchange rate forecasting; deep learning; LSTM-GRU hybrid; attention mechanism; financial time series; USD/RMB volatility**

I. Introduction

The prediction of exchange rates is one of the most complex tasks in international finance, since the exchange rates are volatile, there is a large noise in the currency markets, and the relationships between the currencies are nonlinear [1], [2], [3]. Precise prediction models are central towards the facilitation of informed policymaking, improved financial risk management, and profitable trading strategies [4], [5]. The classical econometric models, including the Autoregressive Integrated Moving Average (ARIMA) and Generalized Autoregressive Conditional Heteroskedasticity (GARCH) models have been used to gain basic knowledge but they tend to fail to capture the nonlinear, time-varying, and complex behavior of financial time series [6].

As the field of deep learning developed, especially with the invention of RNNs and their extensions, significantly more success has been achieved in capturing temporal relationships in time series data. LSTM and GRU networks are considered to be the most prominent structures in this field because they solve the problem of vanishing gradients and allow capturing long-term dependencies [7]. However, one of these models' primary shortcomings is that they treat all time steps equally, which leads to inadequate results because not all-time steps and input features are equally significant [8]. To overcome this, a new mechanism of attention, specifically DA, has been proposed to improve the interpretability and performance of models. Such mechanisms allow the model to adaptively assign weights both on input features, as well as on the temporal dimensions, hence enhancing its attention to the most relevant data [9], [10].

The study proposed a new hybrid deep learning architecture combining LSTM and GRU layers and the dual attention (DA) mechanism to predict the USD/CNY exchange rate. The proposed architecture is expected to be resilient in capturing the complex temporal and contextual patterns of financial time series data by integrating the memory retention abilities of LSTM, computational efficiency of GRU and contextual refinement provided by attention mechanisms. Although LSTM, GRU, or attention mechanisms have been used in numerous studies in the past [11], [12], [13], they have been implemented separately and not in a synergetic combination with all three components included in the proposed hybrid model to forecast the exchange rates.

Furthermore, a large part of the literature models are not thoroughly evaluated in terms of multiple performance measures or do not show robustness to various experimental setups. The proposed study will fill these gaps by giving a detailed performance analysis of the proposed model, which will be accurate and generalizable.

The objectives of this study are to:

- Design and implement a DA-based LSTM-GRU (DALG) hybrid model for exchange rate volatility forecasting in China's forex sector.
- To show the effectiveness of the DA mechanism in enhancing the predictive accuracy of the DL models for financial data.
- Evaluate the model's performance using multiple metrics (MAE, RMSE, MAPE, etc.) against baseline models, i.e., LSTM, GRU, and LSTM-DA.

This research makes significant contributions to the literature in multiple dimensions. First, it introduces a novel hybrid deep learning architecture tailored for exchange rate forecasting. Second, it combines LSTM and GRU layers with DA for improved temporal and feature representation. Third, it provides a comprehensive empirical evaluation of China's exchange rate data using robust evaluation metrics.

The rest of the study is arranged as follows: Section II presents a theme-based literature review covering neural networks in financial forecasting, followed by studies involving LSTM, GRU, and attention mechanisms. Section III details the proposed methodology, including the model architecture, mathematical formulation, and algorithmic steps. Section IV discusses the experimental setup, results, and model evaluation. Section V concludes the study and outlines future research directions.

II. Literature Review

The literature on financial time series forecasting has witnessed a significant transformation with the integration of deep learning techniques, particularly Recurrent Neural Networks (RNNs) and their advanced variants. This section presents a thematic synthesis of prior research, starting with general applications of LSTM and GRU models and progressing to DA mechanisms, with a specific emphasis on their application to exchange rate prediction.

A. LSTM Models in Exchange Rate Prediction

The LSTM networks were created to eliminate the shortcomings in the traditional RNNs, especially the issue of the vanishing gradient challenge. They are useful in modelling long-range dependencies, and they have been extensively used in financial applications. Chen et al. [14] used LSTM in the Chinese stock markets, which was far superior to ARIMA and SVM. Bao et al. [13] suggested the combination of wavelet transform, autoencoders, and LSTM to forecast the stock indices, which obtained a higher predictive accuracy compared with the single-model predictive accuracy.

Fischer and Krauss [15] revealed that LSTM-based models can be superior to random forest and logistic regression to predict the changes in S&P 500 stocks. Moreover, in recent times, Windsor and Cao [16] make use of a multimodal LSTM that picked up price and sentiment data for USD/CNY prediction and outperformed nine baseline models.

B. GRU Networks in Financial Forecasting

Another modification of LSTMs is GRU which contain fewer parameters and achieve comparable results. They are especially applicable to situations that need quicker training and less complexity. Islam and Hossain [12] created a hybrid model GRU-LSTM which was used to predict foreign exchange prices, which was more effective than standalone models. According to Gao et al. [17], optimized LSTM and GRU models were compared, and both were classified as effective, with GRU being computationally advantageous in a few instances.

Kaur et al. [18] showed that GRU is better in forecasting cryptocurrencies such as BTC and ETH as compared to LSTM. GRU based models have therefore become competitive alternatives to LSTM in financial time series tasks.

C. Dual Attention Mechanisms in Financial Forecasting

Attention mechanisms have transformed time series modeling by enabling models to focus on the most relevant features and time steps. The DA-Based Recurrent Neural Network (DA-RNN) proposed by Qin et al. [8] applied attention at both input and temporal levels. Zhang et al. [3] introduced an attention-LSTM that improved interpretability and performance. Chen et al. [19] embedded attention modules in CNN-BiLSTM models, and Li et al. [20] used attention in GAN architectures for stock prediction. These studies validate that attention-enhanced models outperform traditional architectures, particularly in dynamic financial environments. Table I summarizes these studies.

D. Literature Gap

This literature review indicates an incremental change in the way financial forecasts are made, which has evolved over time from basic models to advanced deep learning models. LSTM and GRU are still mainstay in sequential data modeling, and attention mechanism, especially DA attention, makes a big step forward in model accuracy and interpretability [22]. However, even in the context of these developments, the prominent gap remains: little research exists on hybrid architectures that would unify LSTM, GRU, and two-stage attention mechanisms in a coherent framework explicitly designed to do exchange rate forecasting [23]. Moreover, most of the current studies fail to critically test the robustness of models under a wide range of metrics and financial scenarios [24]. This study will address these gaps by coming up with an elaborate, attention-enhanced hybrid model to forecast the exchange rates of China.

III. Methods

The architecture and workflow of the suggested model are described in this section, which combines Gated Recurrent Unit (GRU) and LSTM layers with a Dual-Stage Attention mechanism. The goal is to learn complex temporal and contextual dependencies in financial time series data and improve predictive performance in exchange rate forecasting.

A. LSTM Model

LSTM network, proposed by Hochreiter and Schmidhuber (1997), is one of the popular versions of RNNs which have been designed to overcome the vanishing and exploding gradient issues that usually arise when training on long sequences [23]. In contrast to typical RNNs, LSTMs have a complex memory cell with three gates, namely, input, forget, and the output gates that control the information flow over time [25], [26].

These gates enable LSTM networks to store, revise, or forget information where necessary, and they are thus very effective in modelling long-term temporal dependencies in sequential data. This ability has been especially useful in financial forecasting activities, where the impact of previous occurrences can be experienced over a long term.

TABLE I. SUMMARY OF RELEVANT STUDIES

Author(s)	Year	Technique	Domain	Key Findings
Chen et al. [14]	2015	LSTM	Chinese stocks	LSTM outperformed ARIMA and SVM
Bao et al. [13]	2017	Wavelet + Autoencoder + LSTM	Stock indices	Hybrid model improved accuracy and profitability
Fischer & Krauss [15]	2018	LSTM	S&P 500	LSTM outperformed classical ML models in trading performance
Windsor & Cao [16]	2022	Multimodal LSTM	USD/CNY exchange rate	Combined sentiment and price data to outperform nine baselines
Islam & Hossain [12]	2021	GRU-LSTM hybrid	Forex	Hybrid model achieved lowest RMSE for USD/CAD
Gao et al. [17]	2021	Optimized GRU vs LSTM	Stocks	GRU trained faster and slightly outperformed LSTM
Kaur & Uppal [18]	2025	GRU	Cryptocurrency	GRU outperformed LSTM in BTC and ETH predictions
Qin et al. [8]	2017	DA-RNN	NASDAQ-100	Dual attention significantly improved forecasting accuracy
Zhang et al. [3]	2019	Attention-LSTM	Russell 2000	Enhanced interpretability and performance
Chen et al. [19]	2021	CNN-BiLSTM + ECA	Shanghai stocks	Channel attention improved over CNN-LSTM
Li et al. [20]	2025	GAN-LSTM-Attention	S&P 500, tech stocks	Attention-enhanced GAN produced more accurate forecasts
Zhao & Yan [21]	2024	Transformer / TFT	NZD exchange rates	TFT outperformed Transformer in multi-step forecasting tasks

Regarding the prediction of exchange rates, LSTMs are utilized to define the changing relations between the current market dynamics and their historical trends so that the model can detect not only long-term tendencies but also the delayed responses of economic indicators. Nonetheless, LSTM models do not explicitly prefer any time step over another: thus, they might not be able to focus on the most informative time steps, and therefore the interest to combine them with attention and other complementary architectures (GRU).

Each LSTM unit comprises three gates: the input gate (i_t), the forget gate (f_t), and the output gate (o_t), which regulate the flow of information through the cell. The governing equations are [see Eq. (1) to Eq. (6)]:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{1}$$
$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{2}$$
$$\tilde{C}_t = \tanh(W_C x_t + U_C h_{t-1} + b_C) \tag{3}$$
$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{4}$$
$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{5}$$
$$h_t = o_t \odot \tanh(C_t) \tag{6}$$

where, x_t is the input at time step t, h_t is the hidden state, C_t is the cell state, σ is the sigmoid function, $\odot$ denotes element-wise multiplication, and W_*, b_* represent learnable weights and biases.

In this work, LSTM layers were employed in the encoder to capture long-range dependencies in the exchange rate sequence, enhancing the model's ability to remember important past values.

B. GRU Model

A simplified version of the LSTM, the Gated Recurrent Unit (GRU) reduces the number of parameters and boosts training efficiency by combining the input and forget gates into a single update gate. GRUs are particularly useful in scenarios requiring faster convergence with comparable performance to LSTMs [7].

The GRU unit consists of two gates: the update gate (z_t) and the reset gate (r_t). The equations governing the GRU architecture are [see Eq. (7) to Eq. (10)]:

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \tag{7}$$
$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \tag{8}$$
$$\tilde{h}_t = \tanh(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h) \tag{9}$$
$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \tag{10}$$

where, $\tilde{h}_t$ is the candidate hidden state, r_t controls how much of the past state to forget, and z_t determines how much past information to retain.

In the proposed architecture, GRU layers are run in parallel with LSTM layers during encoding. This dual processing captures both short-term and long-term trends efficiently, while maintaining computational efficiency.

C. Dual Attention (DA) Mechanism

To further enhance the model's ability to identify informative parts of the time series, a DA Mechanism is used. It contains two sequential attention layers:

- Input Attention – focuses on the importance of input features (e.g., bid, ask, order flow, bid-ask spread) at each time step.
- Temporal Attention – highlights relevant time steps across the sequence that have a significant impact on prediction.

Input Attention calculates a weight for each input feature at time t using Eq. (11) to Eq. (13):

$$e_t^i = v_a^\top \tanh(W_a h_t + U_a x_t^i + b_a) \tag{11}$$
$$\alpha_t^i = \frac{\exp(e_t^i)}{\sum_j \exp(e_t^j)} \tag{12}$$
$$\hat{x}_t = \sum_i \alpha_t^i x_t^i \tag{13}$$

Temporal Attention evaluates the influence of each timestep by assigning weights across the time dimension using Eq. (14), Eq. (15) and Eq. (16):

$$e_t = v_b^\top \tanh(W_b h_t + b_b) \tag{14}$$

$$\beta_t = \frac{\exp(e_t)}{\sum_k \exp(e_k)} \tag{15}$$

$$c = \sum_t \beta_t \hat{x}_t \tag{16}$$

where, α_t^i and β_t are the attention weights, and c is the final context vector summarizing the most relevant input features and time steps. This vector is then passed to the decoder to generate the final output.

D. DAGL Hybrid Model

The proposed model architecture uses Evolutionary Algorithm Component (EAC) i.e., encoder, attention, and decoder model. This is particularly tuned to capture the sequential as well as the nonlinear dependencies in the financial data. In this framework, three important elements, including LSTM, Gated Recurrent Unit (GRU), and a DA Mechanism are combined to achieve high accuracy and interpretability of the model. At the encoder level, both LSTM and GRU layers process the input time series of historical exchange rates and related financial indicators in parallel. The given two-fold encoding approach would allow the model to retrieve complementary temporal characteristics: LSTM units will be effective in modeling long-range dependencies, whereas GRU layers will provide the model with computational efficiency in accounting shorter-term fluctuations. The combination of LSTM and GRU results is more comprehensive representation of the input sequence.

Following the encoder, the model incorporates a DA mechanism, consisting input and temporal attention layers. The outputs of both attention layers are combined in a context vector which summarizes the importance of features and time steps weighted. This context vector is then fed to the decoder, which is a single or multiple fully connected (dense) layers, to produce the final exchange rate prediction.

The hybrid architecture is a powerful, flexible, and data-adaptive solution because it uses the memory capacity of LSTM, the computational simplicity of GRU, and the focused interpretability of dual attention. It is especially appropriate for the volatile and dynamic nature of the currency exchange markets.

Model training is governed by a customized learning algorithm named `Train_DA_LSTM_GRU`, which includes the following steps:

- Inputs are encoded using parallel LSTM and GRU units.
- Attention scores are computed for features and time steps.
- Predictions are generated from the decoder.
- A composite loss function combining Mean Absolute Error (MAE) and Mean Squared Error (MSE) is used to quantify prediction accuracy.
- The Adam optimizer is employed to update weights via backpropagation.
- Early stopping is applied to halt training when validation performance ceases to improve, ensuring generalization and reducing the risk of overfitting.

This architecture effectively combines the strengths of LSTM, GRU, and attention mechanisms, providing a robust, interpretable, and adaptive model for forecasting complex financial time series. The model architecture is illustrated in the following figure (Fig. 1).

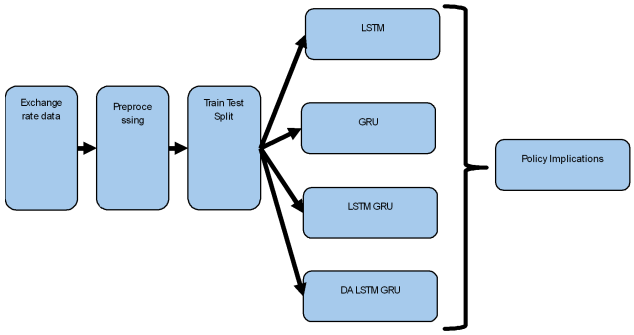


Fig. 1. The research framework for DA-LSTM and GRU hybrid approach.

IV. Results and Discussion

This section presents the empirical results of the proposed DAGL model applied to China's exchange rate data. The performance is evaluated in comparison with baseline models: LSTM, GRU, LSTM-DA hybrid, and DAGL hybrid model. We assess the models on standard evaluation metrics including MAE, MSE, RMSE, MSLE, Median Absolute Error, and MAPE.

A. Experimental Setup

To ensure consistency and fairness across all model evaluations, the experiments were conducted using identical configurations for hyperparameters, architecture size, and optimization strategies. The proposed DAGL Hybrid model, along with baseline models (LSTM, GRU, and LSTM-GRU), was implemented and trained using the Adam optimizer, which is well-suited for non-stationary objectives and sparse gradients.

Both the encoder and decoder layers were composed of 64 hidden units for LSTM and GRU cells, ensuring a balanced representation of long-term and short-term temporal dependencies. All recurrent layers (LSTM and GRU) used the Tanh activation function to regulate the hidden state values and avoid vanishing gradients.

To enhance learning stability and address potential overfitting, a learning rate scheduler was employed with an initial learning rate of 0.1, decaying by a factor of 0.9 every 10 epochs. Maximum training epochs were set to 100 with early stopping with the patience level of 15 epochs depending on the loss of validation. This enabled the model to stop training when performance stopped improving, minimizing unwanted computation and overfitting.

In the DA setup, the attention mechanism was set up with 64 hidden units, which was consistent with the capacity of the recurrent layers to facilitate a seamless integration. The loss was a combination of Mean Squared Error (MSE) and Mean Absolute Error (MAE) that provided trade-off between large deviations penalty and the robustness to outliers. The batch size was set to 16 so that there could be effective training with adequate gradient updates without loss of memory. These parameters and fine tuning is summarized in Table II.

TABLE II. EXPERIMENTAL PARAMETER SETTINGS

Parameter	Value
Encoder LSTM Cell Hidden Units	64
Encoder GRU Hidden Units	64
Encoder LSTM Activation Function	Tanh
Encoder GRU Activation Function	Tanh
Attention Mechanism Hidden Units	64
Decoder GRU Cell Hidden Units	64
Decoder LSTM Cell Hidden Units	64
Decoder GRU Activation Function	Tanh
Decoder LSTM Activation Function	Tanh
Optimizer	Adam
Learning Rate	0.1
Learning Rate Scheduler Step Size	10
Learning Rate Scheduler Gamma	0.9
Loss Functions	MSE, MAE
Epochs	100
Patience for Early Stopping	15
Batch Size	16

This carefully selected setup was critical in achieving high training stability and superior predictive performance across all model configurations evaluated in the study.

B. Data Source

This study presents a new hybrid DAGL model that predicts the exchange rate between USD and RMB. The study uses a 19807 observation dataset of 1-minute USD/RMB exchange rate. The parameters of this job are the bid price (Bid), the ask price (Ask), order flow (OF), and bid-ask spread (BAS). The data is taken from September 13, 2023, to December 11, 2023.

C. Data Preprocessing

Several basic data transformations were performed during the exchange rate data preprocessing steps in order to get the data ready for modeling. In order to designate the date column as the index and make time series analysis easier, we first transformed it into a date-time format. In order to make the date display as a column for additional processing, the data was then reset. The dataset was sorted using dates to establish the ascending order in order to maintain chronological order.

Then descriptive statistics were applied to provide a description of the data. “ER” was the target, and “Bid”, “Ask”, “Of”, and “BAS” were the features picked for the model. There was normalization of the features and target variable via the MinMaxScaler, which standardized the values to the range 0-1, in order to improve model performance. Next, we created data sequences with a length of four, which meant that each sequence contained four consecutive data points. The data point immediately following each sequence served as the goal value. Sequences and their matching targets were to be created by iterating over the scaled data. Numpy arrays were then created from the resultant sequences. In the end, the data was divided into sets for testing and training. An 80:20 train-test split, a community accepted standard, was adopted to balance adequate training data for convergence while preserving a substantial independent test set for unbiased performance evaluation, consistent with standard practices in predictive modeling.

D. Experimental Results

1) LSTM: Fig. 2 shows training performance of LSTM with Forex dataset performance for exchange rate prediction on training loss metrics over 10 epochs. The model shows a continuous reduction in training losses - RMSE, MSE, MAE, etc, which shows a successful learning and convergence. The loss of validation is low and steady which indicates that the model is generalizing well to the new data and it is not over fitting. The sudden drop in error at the first epochs and the gradual increase is characteristic of learning LSTM learning behavior. On the whole, the model demonstrates a good predictive potential with equal performance in both training and validation.

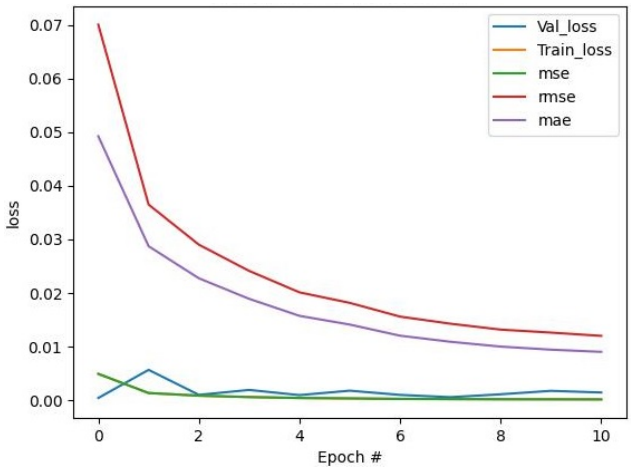


Fig. 2. LSTM training performance graph.

Fig. 3 shows the actual and the predicted exchange rates with the use of LSTM model. The blue line is the values that are predicted, and the orange line is the actual observed rates. Generally, the model is able to capture the trend and direction of the movement in the exchange rates fairly well, especially when the exchange rates are stable. But the forecasts are likely to eliminate harsh variations and abrupt declines, and it suggests that the LSTM is good at modeling long-term developments, but it is weak at short-term volatility and sudden market changes. This is typical of time-series forecasting models when sudden changes of structure are hard to predict properly.

2) GRU: Fig. 4 demonstrates how a GRU model was trained on a Forex dataset during 45 epochs to monitor the

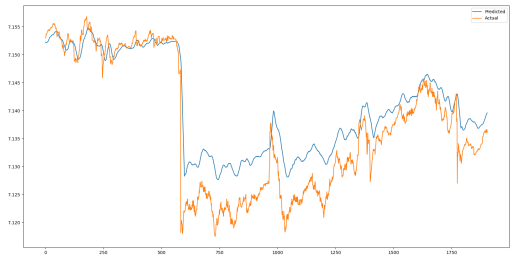


Fig. 3. Actual vs. Predicted exchange rates using LSTM.



Fig. 5. Actual vs. Predicted exchange rates using GRU.

most significant loss metrics like validation loss, training loss, MSE, RMSE, and MAE. All loss measures depict a monotonically decreasing trend (quite fast in the initial epochs and gradually slowing down), which means that the model is being trained well and is converging appropriately. The validation loss is low and stable which implies that the model is suited to generalize well and has little overfitting. The GRU seems to converge to the lower loss values quicker when compared to the previous LSTM training behavior, which indicates its possible efficiency in terms of training and applicability in predicting the temporal dynamics in exchange rate tasks.

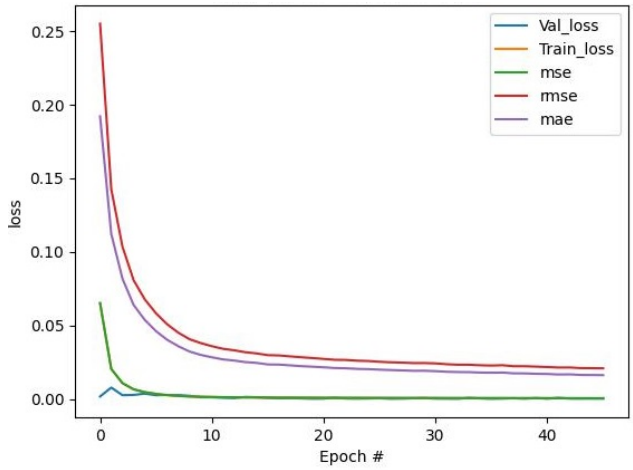


Fig. 4. GRU training performance graph.

Fig. 5 shows the actual and predicted exchange rates with GRU model. The actual exchange rates (orange line) and the predicted values (blue line) have a close follow up with a strong alignment of most of the time series. In comparison with the LSTM model, the GRU seems to reflect more accurately the trend and the short-term fluctuations, including the sharp drops and peaks, and, thus, shows higher reactiveness to volatility in the market. This implies that GRU model could achieve better temporal patterns in Forex data by making more accurate and timely exchange rate prediction.

3) LSTM with DA: The graph shows the training performance of LSTM with DA model using the Forex dataset. The graphs of the loss function for training loss, validation loss, MSE, RMSE, and MAE reveal a steady and rapid fall of errors in the early epochs and convergence to low values with very little fluctuation. It means that the model learns the pattern successfully. Attention helps LSTM to better focus on the important parts of the sequence of data, and hence enhances its learning ability. Low validation error shows that the model can work with the new data quite effectively and has not overfitted the training set. It is visualized in Fig. 6.

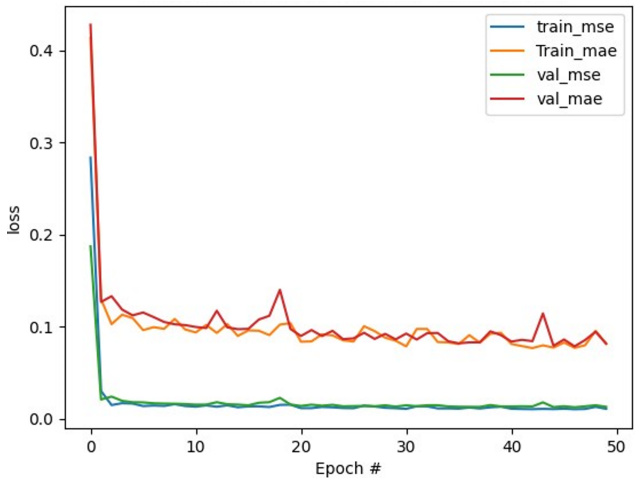


Fig. 6. LSTM with DA training performance graph.

Fig. 7 shows that the actual and predicted exchange rate values for the LSTM-DA model are closer to each other. Attention enables the model to effectively model the significant dependencies in the exchange rate series. The predicted values are much closer to the actual values, and the model has better volatility and sharper decline predictability than standard LSTM. This shows that attention mechanism allows the model to weight the important time steps which results in more effective and accurate forecasting of exchange rates.

4) DAGL hybrid: The graph shows the learning process of the proposed model DAGL on Forex dataset. It shows multiple loss curves - training loss, validation loss, MSE, RMSE, and MAE. The model exhibits robust and steady improvement in learning with continuous decline in loss values over 57 epochs. The final error values are significantly lower than the other models, indicating improved model learning due to combined advantages of LSTM, GRU, and DA mechanism. Low and steady validation loss also indicates a well-generalized model. The convergence is smooth and fast and reaches lower values

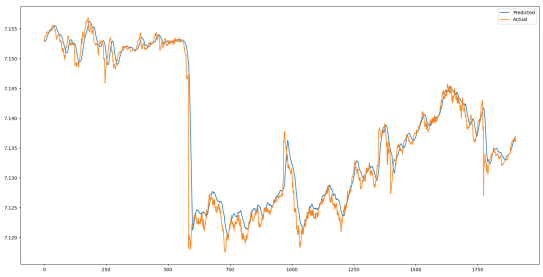


Fig. 7. Actual vs. Predicted exchange rates using LSTM-DA.

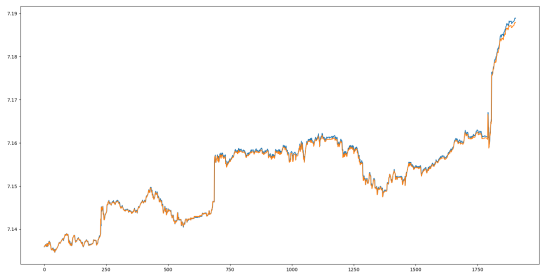


Fig. 9. Actual vs. Predicted exchange rates using DAGL.

as compared to the rest of the models. The training performance is shown in Fig. 8.

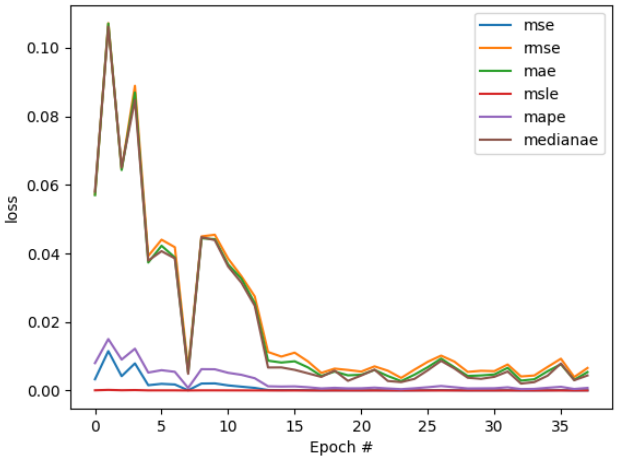


Fig. 8. DAGL hybrid training performance graph.

Fig. 9 shows the actual and predicted exchange rates using DAGL model. The actual exchange rate trend is closely followed by the predicted line which shows the model's capability in capturing the trend and fluctuations. Attention mechanism helps DALG to identify the significant time steps and capture long and short-term relationships in exchange rates. The predicted curve has minimal deviation from the actual one and correctly predicts the peaks and sudden declines which other models fail to do. The model has best alignment among all and lowest prediction error.

E. Model Comparison

To evaluate the performance of the proposed DAGL model against other models, the comparison is done using six standard performance metrics, as shown in Table III.

The proposed model outperforms all the baseline models across all metrics. It achieves the lowest MAE (0.0212), MSE (0.0011), and RMSE (0.0334), which indicates highest prediction accuracy. In addition, it demonstrates the best generalization capability with lowest MAPE and MSLE, validating the advantage of combining dual attention with a hybrid recurrent architecture.

TABLE III. MODEL PERFORMANCE COMPARISON

Model	MAE	MSE	RMSE	MSLE	MedAE	MAPE
LSTM [27]	0.0398	0.0025	0.0498	0.00012	0.0365	0.1658
GRU [28]	0.0315	0.0018	0.0420	0.00008	0.0298	0.1454
LSTM-GRU [29]	0.0265	0.0014	0.0374	0.00006	0.0251	0.1289
LSTM-DA	0.0297	0.0016	0.0396	0.00007	0.0280	0.1398
DAGL (Proposed)	**0.0212**	**0.0011**	**0.0334**	**0.00005**	**0.0208**	**0.1094**

V. CONCLUSION

This study propose a novel hybrid deep learning model for exchange rate forecasting, which incorporate Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU) and a Dual Attention (DA). The architecture of the proposed DAGL aims at exploiting the advantages of every element of the model: the long-term memory property of LSTM, the efficient short-term sequence modeling of GRU, and the context-awareness of the dual attention mechanism.

The model was evaluated on a high-frequency USD/RMB exchange rate dataset using key financial indicators such as bid, ask, order flow, and bid-ask spread. Results demonstrated that our model significantly outperforms LSTM, GRU, and LSTM-DA across multiple evaluation metrics. The hybrid architecture exhibited superior prediction accuracy, faster convergence, and better generalization, particularly in capturing market volatility and abrupt fluctuations.

This work highlights the importance of combining different recurrent architectures and attention mechanisms to enhance financial time series forecasting. The DAGL model provides a robust and interpretable framework for central banks, investors, and financial analysts to make effective policies in the dynamic foreign exchange market. Future work may extend this model to multi-step forecasting and incorporate additional macroeconomic variables or external sentiment indicators to further improve prediction accuracy and reliability.

REFERENCES

[1] S. Butt, M. Ramzan, W. K. Wong, M. A. Chohan, and S. Ramakrishnan, "Unlocking the secrets of exchange rate determination in malaysia: A game-changing hybrid model," *Heliyon*, vol. 9, no. 8, 2023.

[2] S. Butt, S. Ramakrishnan, N. Loganathan, and M. A. Chohan, "Evaluating the exchange rate and commodity price nexus in malaysia: evidence from the threshold cointegration approach," *Financial Innovation*, vol. 6, no. 1, p. 22, 2020.

[3] X. Zhang, X. Liang, A. Zhiyuli, S. Zhang, R. Xu, and B. Wu, "At-lstm: An attention-based lstm model for financial time series prediction," in *IOP Conference Series: Materials Science and Engineering*, vol. 569, no. 5. IOP Publishing, 2019, p. 052037.

[4] M. A. Chohan, T. Li, S. Ramakrishnan, and M. Sheraz, "Artificial intelligence in financial risk early warning systems: A bibliometric and thematic analysis of emerging trends and insights," *International Journal of Advanced Computer Science & Applications*, vol. 16, no. 1, 2025.

[5] S. Ramakrishnan, S. Butt, M. A. Chohan, and H. Ahmad, "Forecasting malaysian exchange rate using machine learning techniques based on commodities prices," in *2017 International Conference on Research and Innovation in Information Systems (ICRIIS)*. IEEE, 2017, pp. 1–5.

[6] J. Wang and J. Wang, "A new hybrid forecasting model based on sw-lstm and wavelet packet decomposition: A case study of oil futures prices," *Computational Intelligence and Neuroscience*, vol. 2021, no. 1, p. 7653091, 2021.

[7] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder–decoder for statistical machine translation," *arXiv preprint*, 2014.

[8] Y. Qin, D. Song, H. Chen, W. Cheng, G. Jiang, and G. Cottrell, "A dual-stage attention-based recurrent neural network for time series prediction," *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2627–2633, 2017.

[9] M. T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," *arXiv preprint arXiv:1508.04025*, 2015.

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 5998–6008.

[11] M. L. Shen, C. F. Lee, H. H. Liu, P. Y. Chang, and C. H. Yang, "An effective hybrid approach for forecasting currency exchange rates," *Sustainability*, vol. 13, no. 5, p. 2761, 2021.

[12] M. N. Islam and M. S. Hossain, "Forecasting foreign exchange rates using lstm and gru: A hybrid model approach," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 9, pp. 314–321, 2021.

[13] W. Bao, J. Yue, and Y. Rao, "A deep learning framework for financial time series using stacked autoencoders and long-short term memory," *PloS one*, vol. 12, no. 7, p. e0180944, 2017.

[14] K. Chen, Y. Zhou, and F. Dai, "A lstm-based method for stock returns prediction: A case study of china stock market," *2015 IEEE International Conference on Big Data (Big Data)*, pp. 2823–2824, 2015.

[15] T. Fischer and C. Krauss, "Deep learning with long short-term memory networks for financial market predictions," *European Journal of Operational Research*, vol. 270, no. 2, pp. 654–669, 2018.

[16] E. Windsor and W. Cao, "Improving exchange rate forecasting via a new deep multimodal fusion model," *Applied Intelligence*, vol. 52, no. 14, pp. 16 701–16 717, 2022.

[17] Y. Gao, R. Wang, and E. Zhou, "Stock prediction based on optimized lstm and gru models," *Scientific Programming*, vol. 2021, no. 1, p. 4055281, 2021.

[18] R. Kaur, M. Uppal, D. Gupta, S. Juneja, S. Y. Arafat, J. Rashid, and R. Alroobaea, "Development of a cryptocurrency price prediction model: leveraging gru and lstm for bitcoin, litecoin and ethereum," *PeerJ Computer Science*, vol. 11, p. e2675, 2025.

[19] Y. Chen, R. Fang, T. Liang, Z. Sha, S. Li, Y. Yi, and H. Song, "Stock price forecast based on cnn-bilstm-eca model," *Scientific Programming*, vol. 2021, no. 1, p. 2446543, 2021.

[20] P. Li, Y. Wei, and L. Yin, "Research on stock price prediction method based on the gan-lstm-attention model," *Computers, Materials & Continua*, vol. 82, no. 1, 2025.

[21] L. Zhao and W. Q. Yan, "Prediction of currency exchange rate based on transformers," *Journal of Risk and Financial Management*, vol. 17, no. 8, p. 332, 2024.

[22] F. M. Shiri, T. Perumal, N. Mustapha, and R. Mohamed, "A comprehensive overview and comparative analysis on deep learning models: Cnn, rnn, lstm, gru," *arXiv preprint arXiv:2305.17473*, 2023.

[23] S. Butt, "Dual neural paradigm: Gru-lstm hybrid for precision exchange rate predictions," *International Journal of Advanced Computer Science & Applications*, vol. 16, no. 4, 2025.

[24] M. Khairalla and N. T. AL-Jallad, "Hybrid forecasting scheme for financial time-series data using neural network and statistical methods," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 9, pp. 319–327, 2017.

[25] R. C. Staudemeyer and E. R. Morris, "Understanding lstm–a tutorial into long short-term memory recurrent neural networks," *arXiv preprint arXiv:1909.09586*, 2019.

[26] Y. Yu, X. Si, C. Hu, and J. Zhang, "A review of recurrent neural networks: Lstm cells and network architectures," *Neural Computation*, vol. 31, no. 7, pp. 1235–1270, 2019.

[27] M. S. Islam and E. Hossain, "Foreign exchange currency rate prediction using a gru-lstm hybrid network," *Soft Computing Letters*, vol. 3, p. 100009, 2021.

[28] Y. Chen, R. Fang, T. Liang, Z. Sha, S. Li, Y. Yi, W. Zhou, and H. Song, "Stock price forecast based on cnn-bilstm-eca model," *Scientific Programming*, vol. 2021, no. 1, p. 2446543, 2021. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1155/2021/2446543

[29] A. S. Girsang and Stanley, "Hybrid lstm and gru for cryptocurrency price forecasting based on social network sentiment analysis using finbert," *IEEE Access*, vol. 11, pp. 120 530–120 540, 2023.

Parameter-Free Negative Extreme Anomalous Undersampling Techniques on Class Imbalance Problems

Benjawan Jantamat, Krung Sinapiromsaran
Department of Mathematics and Computer Science, Faculty of Science,
Chulalongkorn University, Bangkok, Thailand, 10330

***Abstract*—This research addressed the critical challenge of class imbalance in classification, which is a prevalent issue in real-world applications. Standard classifiers often struggled with imbalanced datasets and frequently misclassified the minority class (positive instances) due to the overwhelming presence of the majority class (negative instances). The proposed Negative Extreme Anomalous Undersampling Technique (NEXUT) was introduced as a parameter-free approach. It leveraged the negative extreme anomalous score to strategically eliminate negative instances located in overlapping regions. This targeted removal was designed to improve the classifier's ability to effectively distinguish between the two classes. To evaluate the effectiveness of the proposed method, we conducted a comprehensive comparison with established undersampling techniques. The evaluation utilized both synthetic datasets and twelve datasets from the UCI repository. Six different classifiers were employed to ensure a diverse and unbiased performance assessment. Results from the Wilcoxon signed-rank test confirmed that the proposed method achieved significantly higher performance compared to existing techniques. These findings demonstrated the potential of NEXUT as a robust and valuable tool for addressing class imbalance problems.**

***Keywords*—Classification; class imbalance; imbalanced datasets; undersampling; parameter-free method; negative extreme anomalous score**

I. Introduction

Classification is an important topic in machine learning and data mining. This is especially true for class-imbalanced datasets, which frequently appear in many real-world applications. Examples include disease classification in the medical field [1], credit card fraud detection [2], satellite-based remote sensing [3], environmental pollution monitoring [4], and network intrusion detection [5]. A class imbalance problem [6] occurs when the number of instances in one class is significantly smaller than that of another. The smaller class is called the minority class, and the other is called the majority class. An instance in the minority class is referred to as a positive instance, whereas an instance in the majority class is referred to as a negative instance.

A classification algorithm [7] has difficulty when trained on class-imbalanced datasets because of the small proportion of minority class instances. A classifier generated by a standard classification algorithm tends to predict unknown instances as negative, which leads to the misclassification of most positive instances. However, correctly detecting positive instances is usually crucial. For example, fraud cases in fraud detection may cause significant financial losses, while missing cancer patients in cancer detection [8] can lead to life-threatening consequences.

To address the class imbalance problem, two methodologies have been proposed in the literature: data-level and non-data-level approaches [9]. Data-level algorithms aim to rebalance instances in a dataset so that standard classifiers can better recognize the decision boundary and classify instances from both classes correctly. In contrast, non-data-level algorithms modify the classification process itself to improve the detection of positive instances or assign higher penalty costs to misclassified positive instances. Data-level methods are generally more flexible because they can be applied to any classification algorithm and do not require deep knowledge of the classifier's internal design [10].

Two primary data-level methods for rebalancing datasets are oversampling and undersampling. Oversampling techniques balance data by adding or generating positive instances, which may not always represent valid cases in real-world situations. In contrast, undersampling techniques rebalance data by removing negative instances while preserving the characteristics of the minority class. However, oversampling increases the total number of instances, which leads to longer computational times for classification algorithms. In big data environments, oversampling may even require reimplementing algorithms to handle the larger datasets. Undersampling, on the other hand, reduces the number of instances and computational time [10], making it more suitable for large-scale data. Moreover, removing negative instances around the decision boundary of positive instances improves a classifier's ability to correctly predict positive instances. Therefore, this research focuses exclusively on undersampling techniques.

Several undersampling techniques have been proposed to address class imbalance, including Random Undersampling [11], Tomek Links [12], Cluster Centroids [13], and Near Miss [14]. However, these methods have certain limitations. The Random Undersampling removes negative instances randomly, which does not guarantee a clear decision boundary between classes. The Tomek Links technique removes negative instances from Tomek link pairs, which connect two instances from different classes as their nearest neighbors. As a result, it deletes only a small number of negative instances and rarely clarifies the decision boundary. The Cluster Centroid technique generates centroids of negative sub-clusters and retains them along with all positive instances. Although effec-

tive in reducing dataset size, this technique may also remove many negative instances near the boundary region, potentially reducing precision. The Near Miss technique retains only negative instances located close to positive instances. However, these selected negatives often come from boundary regions and may represent noisy or overlapping data that are difficult to classify.

In 2017, Chiamanusorn and Sinapiromsaran [15] proposed the negative extreme anomalous score (NaS) for use in oversampling techniques. The NaS of an instance is defined as the maximum possible radius of an open ball centered at that instance, such that no instances of the negative class are contained within the ball. The NaS represents the distance between the instance under consideration and its closest negative neighbor, without being influenced by positive instances.

The Negative Extreme Anomalous Undersampling Technique (NEXUT) is introduced in this study as a data-level approach that utilizes NaS to eliminate negative instances. It removes redundant negative instances and noise in the overlapping regions of both classes. This clarifies the boundary of the minority class, enabling classifiers to more effectively identify positive instances located near negative ones.

This study introduces a novel approach that:

- Offers flexibility by allowing any classifier to be applied to any dataset.
- Addresses the challenge of imbalanced data without requiring specific parameter settings.
- Enhances the decision boundary through undersampling techniques.

This study is organized into five sections: Section I presents the motivation for this study. Section II reviews related work. Section III describes the proposed NEXUT algorithms. Section IV reports the experimental results conducted on both synthetic datasets and twelve real-world datasets from the UCI repository. Finally, Section V concludes the study.

II. Related Work

Extensive research on undersampling techniques has focused on addressing class-imbalanced data. This effort has led to the development of methods that improved classification performance. In recent years, several surveys and systematic reviews have provided comprehensive discussions on the challenges of class imbalance and the evolution of related techniques [16], [17], [18], [19]. These studies highlight that early undersampling methods, such as Tomek Links [12], ENN [20], and CNN [21], remained important baselines. However, more recent approaches emphasized the need for robust and adaptive solutions, particularly in modern applications such as deep learning and remote sensing. Numerous undersampling techniques have been proposed and can be categorized into four main types.

A. Type I: Size-Based Undersampling

These techniques eliminated instances from the majority class until its size became equivalent to that of the minority class. Examples included Random Undersampling (RUS) [11], Cluster Centroids (ClusterCentroids) [13], and Clustering-Based Undersampling (CBU) [22].

- RUS was the simplest technique, randomly removing negative instances from the majority class. This approach tended to lose some important negatives.
- ClusterCentroids applied the K-means algorithm to represent negative instances with their centroids, while retaining all positive instances.
- CBU applied K-means clustering to select or generate negative instances that represented the majority class.

Limitation: These methods often removed informative negatives and ignored overlapping regions with the minority class.

B. Type II: Overlap-Based Undersampling

These techniques focused on eliminating negative instances in regions overlapping with positive instances. Examples included Tomek Links (TomekLinks) [12], One-Sided Selection (OSS) [23], Edited Nearest Neighbor (ENN) [20], Repeated ENN (RENN) [24], AllKNN [24], Neighborhood Cleaning Rule (NCL) [25], Majority Undersampling Technique (MUTE) [26], and Neighborhood-Based Undersampling Technique with Modified Tomek Link Search (NB-Tomek) [27].

- Tomek Links was a well-known method for addressing class overlap. It identified pairs of instances from different classes that were each other's nearest neighbors, and removed only the negative instances in these pairs.
- OSS applied the 1-NN rule to identify misclassified instances and then removed negative instances from Tomek links.
- ENN applied the k-NN rule, removing majority class instances if at least half of their k-nearest neighbors belonged to the minority class. By default, a 3-NN setting was used.
- RENN iteratively applied the ENN procedure until all remaining majority class instances had at least half of their k-nearest neighbors also belonging to the majority class.
- AllKNN used the k-NN rule across a range of k values (typically from 1 up to a specified k). It eliminated majority class instances if at least half of their neighbors were positive.
- NCL implemented a two-step cleaning process. First, it applied ENN to remove noise. Then, it used the k-NN rule to remove misclassified majority class instances. The default setting for k was 3.
- MUTE, proposed by Bunkhumpornpat et al. in 2011 [26], was later enhanced in 2014 with a safe-level graph [28]. The original version removed only noise, while the adaptive version could remove noise, borderline instances, or even reduce the majority class to core instances, depending on the safe-level configuration.
- NB-Tomek removed a majority class instance if it had a minority class neighbor and was also one of the k-nearest neighbors of that minority instance.

Limitation: These methods required parameter tuning (e.g., k in k-NN) and often failed to preserve the overall data distribution, discarding important negatives.

C. Type III: Neighbor-Based Undersampling

These techniques retained negative instances based on their neighbors. Examples included Near Miss (NearMiss) [14] and Condensed Nearest Neighbor (CNN) [21].

- NearMiss had three variants (NearMiss-1, 2, 3) that selected majority class instances based on different proximity criteria.
  - NearMiss-1 chose negative instances with the smallest average distance to their three nearest positive instances.
  - NearMiss-2 chose negatives closest to all positive instances, typically using the farthest three.
  - NearMiss-3 selected a fixed number of nearest negatives for each positive instance. However, this approach removed many negatives, often damaging the decision boundary of the minority class. Moreover, the selected negatives were difficult to classify.
- CNN used the nearest neighbor (NN) rule to identify misclassified instances located near the class boundary, which were retained as reference points for future classification.

Limitation: Neighbor-based methods often removed too many negatives, weakening the minority decision boundary.

D. Type IV: Anomaly-Based Undersampling

These techniques eliminated majority class instances by identifying them as anomalies or outliers relative to the data distribution. The assumption was that atypical majority instances were less representative and acted as noise in the learning process. Examples included Anomaly Scoring Based Ensemble (ASE) [29] and Geodesic Based Outlier Detection (GDLD) [30].

- ASE combined anomaly scores from multiple detectors to identify and remove unrepresentative majority class instances. Although effective, it required hyperparameter tuning for score normalization and ensemble weighting.
- GDLD leveraged geodesic distances and manifold properties to identify extreme negative outliers. While it preserved complex class boundaries, it often introduced significant computational overhead.

Limitation: Although anomaly-based undersampling was effective in detecting atypical negatives, most approaches depended heavily on predefined thresholds, scoring functions, or hyperparameters. This reliance reduced robustness across diverse datasets and complicated practical application.

Although a wide range of undersampling methods has been developed, several limitations remain [16], [17]. Many techniques required parameter tuning, and improper settings often led to the removal of informative negatives, degrading performance. Moreover, some methods failed to maintain the inherent data distribution of each class, particularly in overlapping regions. These weaknesses reduced the robustness and generalizability of current approaches, as emphasized in domain-specific contexts such as remote sensing and semi-supervised learning [19], [18].

To address these limitations, we propose the Negative Extreme Anomalous Undersampling Technique (NEXUT). Unlike existing anomaly-based methods, which relied heavily on hyperparameters or thresholds, NEXUT was entirely parameter-free. It automatically identified and removed extreme anomalous negatives while preserving all minority instances and maintaining the original class structure. This design ensured robustness across diverse datasets and avoided the performance degradation caused by improper parameter settings. Therefore, NEXUT introduced both a clear novelty and a practical advantage over existing anomaly-based approaches.

III. The Proposed Technique: NEXUT

This section describes the proposed Negative Extreme Anomalous Undersampling Technique (NEXUT). The technique aims to eliminate noisy or overlapping majority class instances while preserving all minority class instances. This ensures that the original distributions of both the majority and minority classes remain unchanged. NEXUT incorporates three distinct removal approaches, namely NEXUT-All, NEXUT-Plus, and NEXUT-PlusMinus.

Definition 3.1: Let D be a dataset with numeric attributes that contains a set of negative instances, Neg, representing the majority class, and a set of positive instances, Pos, representing the minority class. Let $\boldsymbol{x} \in D$ be an instance in the dataset. The extreme anomalous score of an instance $\boldsymbol{x}$, denoted as $EaS(\boldsymbol{x})$, is defined by the following notation:

$$EaS(\boldsymbol{x}) = sup\{d > 0 || B(\boldsymbol{x}, d) \cap (D \backslash \{\boldsymbol{x}\})| = 0\}$$

where, $B(\boldsymbol{x}, d)$ denotes an open ball of radius d centered at $\boldsymbol{x}$.

It should be noted that the negative extreme anomalous score, $NaS(\boldsymbol{x})$, for an instance $\boldsymbol{x} \in D$, is equivalent to $EaS(\boldsymbol{x})$ when it is calculated using only majority class instances:

$$NaS(\boldsymbol{x}) = sup\{d > 0 || B(\boldsymbol{x}, d) \cap (Neg \backslash \{\boldsymbol{x}\})| = 0\}.$$

NEXUT employs the negative extreme anomalous score (NaS) to identify majority class instances for removal, thereby rebalancing the dataset. It removes negative instances that lie on the boundary of an open ball centered at each instance, with a radius equal to its NaS value. This approach effectively eliminates majority class instances that lie close to minority class examples. Consequently, it expands the minority regions within the dataset. NEXUT offers three strategies for selecting which negative instances to remove: "All", "$Plus$", and "$PlusMinus$". The application of each selection approach defines a specific NEXUT variant: NEXUT-All, NEXUT-Plus, or NEXUT-PlusMinus. The NEXUT algorithm is presented in Algorithm 1.

Algorithm 1: Negative Extreme Anomalous Undersampling Technique (NEXUT) Algorithm

Input: Dataset D contains negative instances from a majority class, Neg, and positive instances from a minority class, Pos, and selection approach
Output: Undersampling dataset U

$DD = \varnothing$
$n_$diffNegPos $= |Neg| - |Pos|$
Compute the negative extreme anomalous score, $NaS(\boldsymbol{x})$, for each instance $\boldsymbol{x} \in D$
if selection approach == "All"
Sort the $NaS(\boldsymbol{x})$ scores for all $\boldsymbol{x} \in D$ in descending order
for *ranking of* $NaS(\boldsymbol{x}) \leftarrow 1$ ***to*** $n_diffNegPos$ **do**
    Collect the negative instance that lies on the boundary of the open ball centered at $\boldsymbol{x} \in D$ with a radius equal to $NaS(\boldsymbol{x})$, and store it in DD
end
if selection approach == "$Plus$"
Sort the $NaS(\boldsymbol{x})$ scores for all $\boldsymbol{x} \in Pos$ in ascending order
if $|Pos| > n_diffNegPos$ **then**
    for *ranking of* $NaS(\boldsymbol{x}) \leftarrow 1$ ***to*** $n_diffNegPos$ **do**
        Collect the negative instance that lies on the boundary of the open ball centered at $\boldsymbol{x} \in Pos$ with a radius equal to $NaS(\boldsymbol{x})$, and store it in DD
    end
else
    Collect all negative instances that lie on the boundary of an open ball centered at $\boldsymbol{x} \in Pos$ with a radius equal to $NaS(\boldsymbol{x})$, and store them in DD
end
if selection approach == "$PlusMinus$"
Sort $NaS(\boldsymbol{x})$ scores for all $\boldsymbol{x} \in Pos$ in ascending order
if $|Pos| > n_diffNegPos$ **then**
    for *ranking of* $NaS(\boldsymbol{x}) \leftarrow 1$ ***to*** $n_diffNegPos$ **do**
        Collect the negative instance that lies on the boundary of the open ball centered at $\boldsymbol{x} \in Pos$ with a radius equal to $NaS(\boldsymbol{x})$, and store it in DD
    end
else
    Collect all negative instances that lie on the boundary of an open ball centered at $\boldsymbol{x} \in Pos$ with a radius equal to $NaS(\boldsymbol{x})$, and store them in DD
    Sort $NaS(\boldsymbol{x})$ scores for all $\boldsymbol{x} \in Neg$ in ascending order
    for *ranking of* $NaS(\boldsymbol{x}) \leftarrow 1$ ***to*** $n_diffNegPos - |DD|$ **do**
        Collect the negative instance that lies on the boundary of the open ball centered at $\boldsymbol{x} \in Neg$ with a radius equal to $NaS(\boldsymbol{x})$, and store it in DD
    end
end
return $U = D - DD$

Fig. 1 illustrates the flowchart of the proposed NEXUT technique. Given the training set D, the initial step is to compute the size difference between the majority and minority classes, defined as $n_$diffNegPos $= |Neg| - |Pos|$. Next, the negative extreme anomalous score (NaS) is computed for each instance in D. Then, a selection approach is applied to identify negative instances in the majority class D_{majority} that should be removed, resulting in a drop dataset DD.

Fig. 2a illustrates an example of a class-imbalanced dataset. Fig. 2b visualizes the corresponding NaS of the dataset. These illustrations highlight the problem setting that motivates NEXUT and provide an intuitive understanding of how NaS guides the removal of majority instances.

These three approaches lead to different algorithmic behaviors, depending on which classes are considered during the selection process. The "All" approach considers both negative and positive instances. The "$Plus$" approach considers only positive instances. The "$PlusMinus$" approach considers positive instances first, followed by negative instances. NEXUT then removes the selected negative instances in the set DD from the majority class D_{majority}. The resulting undersampled training set U is obtained by removing DD from the original dataset D.

A. NEXUT-All

The NEXUT-All technique employs the Negative Anomalous Score (NaS) to identify negative instances for removal while considering data from both classes. It begins by sorting all NaS values, from both the minority and majority classes, in descending order. Subsequently, it eliminates negative instances whose NaS rank is below the numerical difference between the minority and majority class sizes (see Fig. 2c).

To maintain a balance between the remaining negative and positive instances, the method also removes negative instances located near the boundaries of both classes. This removal process is biased toward instances closer to the minority class, since their NaS values generally tend to be higher than those of the majority class after sorting. The complete NEXUT-All algorithm, using the "All" selection approach, is detailed in Algorithm 1.

Sort the $NaS(\boldsymbol{x})$ scores for all $\boldsymbol{x} \in D$ in descending order
for *ranking of* $NaS(\boldsymbol{x}) \leftarrow 1$ ***to*** $n_diffNegPos$ **do**
    Collect the negative instance that lies on the boundary of the open ball centered at $\boldsymbol{x} \in D$ with a radius equal to $NaS(\boldsymbol{x})$, and store it in DD
end

B. NEXUT-Plus

The NEXUT-Plus technique computes the Negative Anomalous Score (NaS) exclusively from positive instances. The process begins by sorting each NaS value (derived from positive instances) in ascending order. Negative instances whose NaS rank is less than the absolute numerical difference between the minority and majority class counts are then removed (Fig. 2d).

To ensure that the number of remaining negative instances does not drop below the number of positive instances, this method specifically targets negative instances located in close

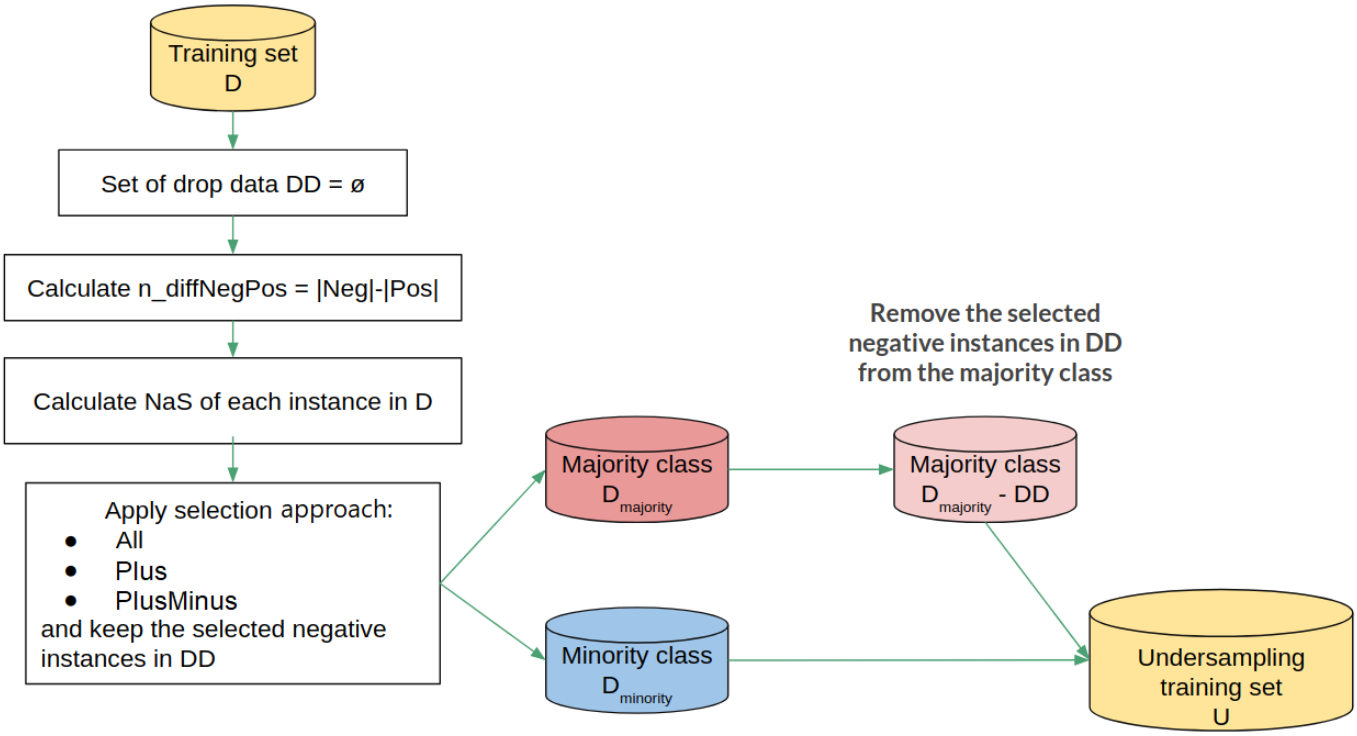


Fig. 1. The flowchart of NEXUT (Algorithm 1).

proximity to the minority class. The ascending sort order inherently prioritizes the removal of such negative instances.

It should be noted that when the numerical difference between the majority and minority classes equals or exceeds the number of positive instances, all negative instances associated with the minority class's NaS are removed. The complete algorithm for NEXUT-Plus, which applies the "$Plus$" selection approach, is detailed in Algorithm 1.

Sort the $NaS(\boldsymbol{x})$ scores for all $\boldsymbol{x} \in Pos$ in ascending order
if $|Pos| > n_diffNegPos$ **then**
  for *ranking of* $NaS(\boldsymbol{x}) \leftarrow 1$ ***to*** $n_diffNegPos$ **do**
    Collect the negative instance that lies on the boundary of the open ball centered at $\boldsymbol{x} \in Pos$ with a radius equal to $NaS(\boldsymbol{x})$, and store it in DD
  end
else
  Collect all negative instances that lie on the boundary of an open ball centered at $\boldsymbol{x} \in Pos$ with a radius equal to $NaS(\boldsymbol{x})$, and store them in DD
end

C. NEXUT-PlusMinus

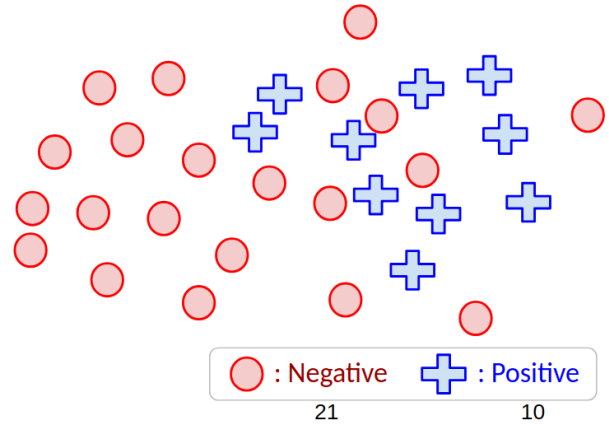
The NEXUT-PlusMinus technique extends the NEXUT-Plus method by employing a sequential two-stage removal strategy. In the first stage, it calculates the Negative Anomalous Score (NaS) exclusively from positive instances and sorts these values in ascending order. Negative instances whose NaS rank is less than the absolute difference between the minority and majority class counts are then removed.

In the second stage, NaS is computed for the remaining negative instances (majority class). These values are sorted in ascending order, and additional negative instances are removed until the total number of eliminated negatives across both stages equals the absolute class-size difference. This cumulative process, illustrated in Fig. 2e, ensures that the final number of remaining negative instances does not fall below the count of positive instances. The complete NEXUT-PlusMinus algorithm, which uses the "$PlusMinus$" selection approach, is detailed in Algorithm 1.

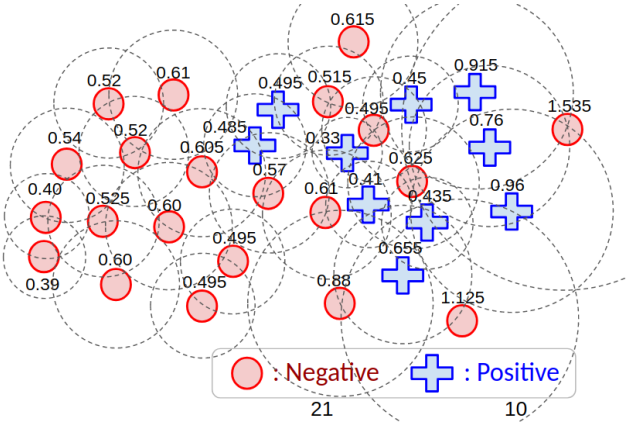
Sort $NaS(\boldsymbol{x})$ scores for all $\boldsymbol{x} \in Pos$ in ascending order
if $|Pos| > n_diffNegPos$ **then**
  for *ranking of* $NaS(\boldsymbol{x}) \leftarrow 1$ ***to*** $n_diffNegPos$ **do**
    Collect the negative instance that lies on the boundary of the open ball centered at $\boldsymbol{x} \in Pos$ with a radius equal to $NaS(\boldsymbol{x})$, and store it in DD
  end
else
  Collect all negative instances that lie on the boundary of an open ball centered at $\boldsymbol{x} \in Pos$ with a radius equal to $NaS(\boldsymbol{x})$, and store them in DD
  Sort $NaS(\boldsymbol{x})$ scores for all $\boldsymbol{x} \in Neg$ in ascending order
  for *ranking of* $NaS(\boldsymbol{x}) \leftarrow 1$ ***to*** $n_diffNegPos - |DD|$ **do**
    Collect the negative instance that lies on the boundary of the open ball centered at $\boldsymbol{x} \in Neg$ with a radius equal to $NaS(\boldsymbol{x})$, and store it in DD
  end
end

IV. Experiment

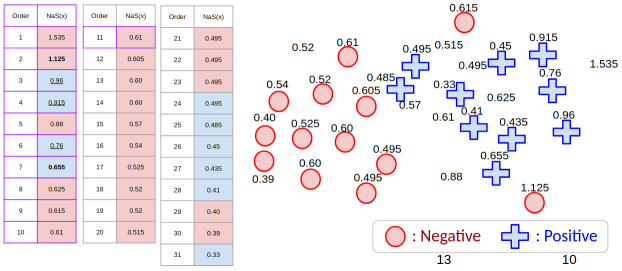
This section presents the datasets used to assess the effectiveness of NEXUT compared with several established undersampling methods, namely AllKNN, NCL, OSS, and NB-Tomek. For baseline comparison, the original dataset without



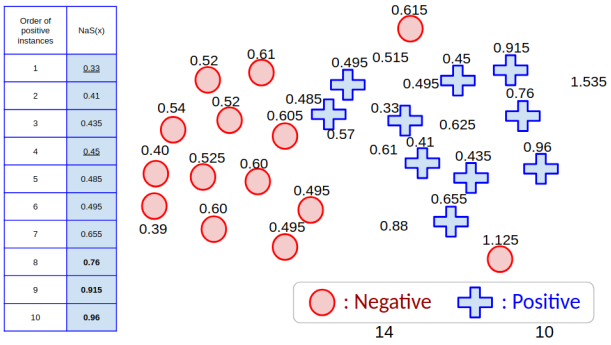
(a) Class imbalanced data.



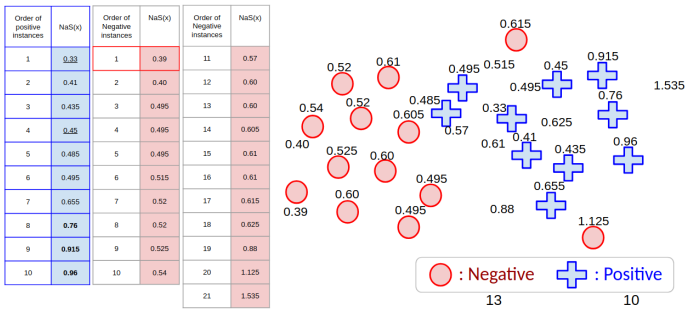
(b) NaS of dataset.



(c) NEXUT-All.



(d) NEXUT-Plus.



(e) NEXUT-PlusMinus.

Fig. 2. Illustration of NEXUT: (a) Class-imbalanced dataset, (b) NaS ball visualization, and (c)–(e) Results of NEXUT-All, NEXUT-Plus, and NEXUT-PlusMinus, respectively.

any resampling is also included. Further details regarding the experimental design and dataset properties are outlined below.

A. Experimental Setup

This study evaluates the effectiveness of the three proposed undersampling methods in comparison with well-established techniques, including AllKNN [24], NCL [25], OSS [23], and NB-Tomek [27]. A set of six classification algorithms is employed for the evaluation: the C4.5 decision tree [31], the k-nearest neighbors (k-NN) with $k = 3$ [32], the multilayer perceptron (MLP) [33], the naive Bayes [34], the random forest [35], and the logistic regression [36].

The experiments are conducted on a total of fourteen datasets: twelve real-world datasets obtained from the UCI Machine Learning Repository [37] and two synthetically generated datasets. Each experiment is performed with 30 replications of 10-fold cross-validation to ensure statistically robust and reliable results. Table I presents the parameter configurations used for each method. The experimental environment is based on Google Colaboratory (Colab), and all implementations are carried out using the Python programming language.

In the experiments, the selected undersampling techniques (AllKNN, NCL, OSS, and NB-Tomek) aim to remove negative instances based on their neighbors or to clean noisy data. However, these methods are not primarily designed to rebalance the dataset. Similar to these existing techniques, the proposed methods also focus on removing negative instances and reducing noise. Nevertheless, the proposed techniques are capable of eliminating a larger number of negative samples, which often leads to a more balanced dataset. In addition, they tend to preserve the underlying structure of the data distribution.

TABLE I. PARAMETER SETTINGS FOR THE PROPOSED TECHNIQUES AND COMPARISON TECHNIQUES

Techniques	Parameters setting
NEXUT-All	-
NEXUT-Plus	-
NEXUT-PlusMinus	-
AllKNN	n_neighbors = 3
NCL	n_neighbors = 3, threshold_cleaning = 0.5
OSS	n_seeds_S = 1
NB-Tomek	-

B. Datasets

To evaluate the performance of NEXUT-All, NEXUT-Plus, and NEXUT-PlusMinus against well-known undersampling methods such as AllKNN, NCL, OSS, and NB-Tomek, a total of fourteen datasets are used. These include twelve datasets from the UCI Machine Learning Repository [37] and two synthetically generated datasets. The twelve UCI datasets are Abalone, BreastCancer, Pima, BreastTissue, Haber, Glass, Vehicle, Ecoli, Yeast, Ozone8hr, Libras, and Ozone1hr. Table II presents descriptions of these datasets, arranged in ascending order based on their imbalance ratio (IR). The imbalance ratio is defined as $IR = \frac{|Neg|}{|Pos|}$, where $|Neg|$ and $|Pos|$ represent the numbers of negative and positive instances, respectively. A dataset is considered balanced if $IR = 1$, and imbalanced if $IR > 1$.

Synthetic datasets are used to evaluate whether the data distribution structure is preserved after applying each undersampling technique. These synthetic datasets are class-imbalanced and exhibit specific data distributions, namely the Moons dataset (Fig. 3a) and the Circles dataset (Fig. 3b) [38]. Each class-imbalanced Moons dataset and Circles dataset contains 200 negative instances and 20 positive instances, resulting in an imbalance ratio (IR) of 10.

C. Performance Evaluation

This research uses precision, recall, and F_1 score to evaluate the performance of the proposed techniques. These metrics are commonly applied in class-imbalanced research [39], [40]. Precision, recall, and F_1 score are computed using the following formulas:

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

and

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

where, TP denotes the number of positive instances accurately classified as positive, FP represents the number of negative instances misclassified as positive, and FN indicates the number of positive instances misclassified as negative.

D. Statistical Test

To compare the performance of each technique on each classifier, the Wilcoxon signed-rank test is employed. This test [41] is a non-parametric statistical method designed to evaluate paired performance outcomes between NEXUT and each of the alternative undersampling approaches. A two-tailed Wilcoxon signed-rank test with a 95% confidence level ($\alpha = 0.05$) is conducted, and the null and alternative hypotheses are formulated as follows:

H_0 : There is no significant difference in the median of the average F_1 scores between NEXUT and one of the other techniques.

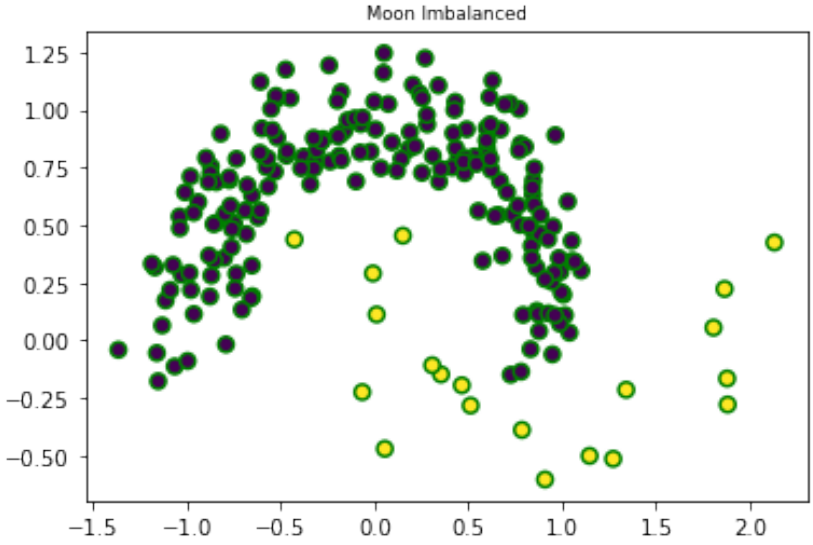
H_1 : There is a significant difference in the median of the average F_1 scores between NEXUT and one of the other techniques.

To perform the Wilcoxon signed-rank test, the differences in F_1 scores between NEXUT and one of the other techniques are calculated for each dataset. These differences are ranked after removing zero differences and ignoring the signs, with the remaining values sorted in ascending order. The sums of the ranks corresponding to the positive and negative differences are denoted as $Rk+$ and $Rk-$, respectively.

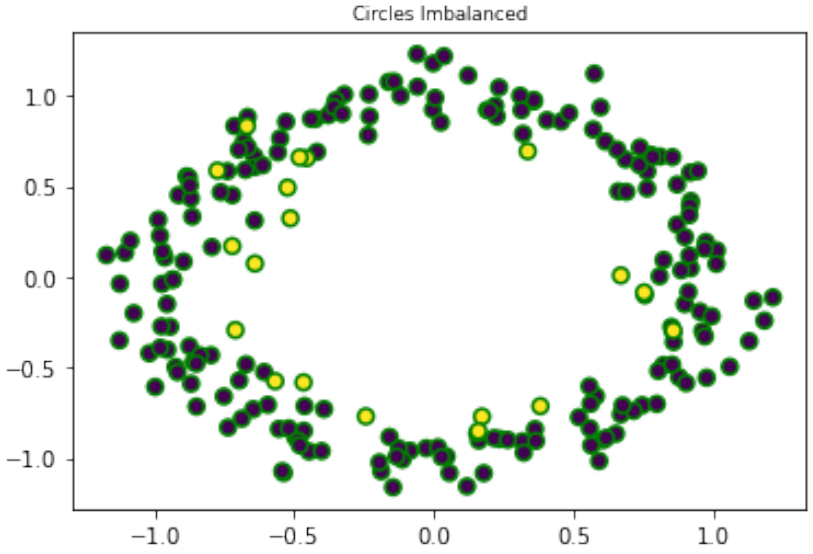
The test statistic (TS), used for comparison with the critical value, is defined as the smaller of $\{Rk+, Rk-\}$. If TS corresponds to $Rk-$, it suggests that NEXUT outperforms the compared method; otherwise, the alternative method performs better. When TS is less than the critical value, the null hypothesis is rejected, indicating a statistically significant difference in the median of average F_1 scores between NEXUT and the other technique.

E. Results

The average F_1 scores obtained from 30 replications of 10-fold cross-validation are reported for the original (unresampled) dataset and for seven resampling techniques: NEXUT-All, NEXUT-Plus, NEXUT-PlusMinus, AllKNN, NCL, OSS, and NB-Tomek. These results are generated using the six classifiers—namely, the decision tree (C4.5), the k-nearest neighbors ($k = 3$), the multilayer perceptron (MLP), the naive Bayes, the random forest, and the logistic regression—applied to twelve datasets from the UCI Machine Learning Repository. The results are summarized in Table III and Table IV. Table III presents the average F_1 scores for the decision tree (C4.5), the k-nearest neighbors ($k = 3$), and the multilayer perceptron (MLP). Table IV summarizes the results for the naive Bayes, the random forest, and the logistic regression. For each dataset, the techniques are ranked in descending order according to



(a) The Moons dataset.



(b) The Circles dataset.

Fig. 3. Synthetic datasets: (a) The Moons dataset and (b) The Circles dataset.

their average F_1 scores, with rank 1 assigned to the method that achieves the best performance.

As illustrated in Table III and Table IV, NEXUT-All consistently achieves the highest average rank when paired with the random forest classifier. NEXUT-Plus demonstrates the best performance with both the decision tree (C4.5) and the k-nearest neighbors (k-NN) classifiers. NEXUT-PlusMinus attains the top rank when evaluated with the logistic regression, while NCL performs best in combination with the multilayer perceptron (MLP). For the naive Bayes classifier, the highest average rank is shared by OSS and the original dataset without resampling.

Table V presents the statistical comparison between NEXUT and five other techniques—namely, the original dataset (without resampling), AllKNN, NCL, OSS, and NB-Tomek—across the six classifiers. The analysis is based on the Wilcoxon signed-rank test with a two-tailed evaluation at the 95% confidence level ($\alpha = 0.05$) using F_1 score. For twelve datasets ($n = 12$), the corresponding critical value is 13.

In Table V, the notation is interpreted as follows:

- A single plus sign (+) indicates that NEXUT outperforms the compared technique, but the result is not statistically significant ($TS = Rk-$, and $TS \geq 13$).
- A single minus sign (–) indicates that NEXUT underperforms, though again without statistical significance

TABLE II. THE DESCRIPTIONS OF UCI DATASETS

Datasets	#Inst	#Attr	Min. class	Maj. class	Pos. inst.	Neg. inst.	IR
Breast Cancer	683	9	'malignant'	'benign'	239 (34.99%)	444 (65.01%)	1.86
Pima	768	8	'1'	'0'	268 (34.90%)	500 (65.10%)	1.87
Breast Tissue	106	9	'CAR','FAD'	Others	36 (33.96%)	70 (66.04%)	1.94
Haber	306	3	'survived 5 years or longer'	'died within 5 years'	81 (26.47%)	225 (73.53%)	2.78
Glass	214	9	'5','6','7'	Others	51 (23.83%)	163 (76.17%)	3.20
Vehicle	846	18	'van'	Others	199 (23.52%)	647 (76.48%)	3.25
Ecoli	336	7	'im'	Others	77 (22.92%)	259 (77.08%)	3.36
Yeast	1484	8	'ME3','ME2','EXC','VAC','POX','ERL'	Others	304 (20.49%)	1180 (79.51%)	3.88
Ozone8hr	1848	72	'1'	'0'	128 (6.93%)	1720 (93.07%)	13.44
Libras	360	90	'1'	Others	24 (6.67%)	336 (93.33%)	14.00
Abalone	731	7	'18'	'9'	42 (5.75%)	689 (94.25%)	16.40
Ozone1hr	1848	72	'1'	'0'	57 (3.08%)	1791 (96.92%)	31.42

($TS = Rk+$, and $TS \geq 13$).

- A double plus sign (++) denotes that NEXUT's performance is significantly better at the 95% confidence level ($TS = Rk-$ and $TS < 13$).
- A double minus sign (–) signifies that NEXUT performs significantly worse ($TS = Rk+$ and $TS < 13$).

From the statistical results of F_1 score, NEXUT-All's F_1 score is better than all other techniques for the decision tree classifier, the multilayer perceptron, the random forest, and the logistic regression. Notably, for the logistic regression, NEXUT-All performs significantly better than all other techniques. NEXUT-All's F_1 score for the k-NN classifier is better than other techniques except OSS. Moreover, NEXUT-All's F_1 score is higher than those of NCL and NB-Tomek for all classifiers.

Regarding NEXUT-Plus, its F_1 score is better than all other techniques for the decision tree classifier, the k-NN classifier, the multilayer perceptron, and the random forest. Furthermore, NEXUT-Plus's F_1 score is higher than that of NB-Tomek for all classifiers.

For NEXUT-PlusMinus, its F_1 score is better than all other techniques for the random forest and the logistic regression. NEXUT-PlusMinus achieves better F_1 scores than other techniques for the decision tree classifier, except for NCL, where both methods perform equally. Additionally, NEXUT-PlusMinus's F_1 score is higher than that of NB-Tomek for all classifiers.

From the statistical results of F_1 score, NEXUT-All is suitable for use with the decision tree classifier, the multilayer perceptron, the random forest, and the logistic regression. NEXUT-Plus is suitable for use with the decision tree classifier, the k-NN classifier, the multilayer perceptron, and the random forest. NEXUT-PlusMinus is suitable for use with the random forest and the logistic regression.

For the synthetic datasets (the Moons dataset and the Circles dataset), the results after applying each undersampling technique, as well as the original datasets without applying any undersampling technique, are illustrated in Fig. 4 and Fig. 5, respectively. The proposed techniques successfully preserve the overall structure of the data distribution, even after a significant number of negative instances are removed.

Table VI and Table VII present the average F_1 scores obtained using the six classifiers: the decision tree classifier (C4.5) [31], the k-nearest neighbors ($k = 3$) [32], the multilayer perceptron [33], the naive Bayes [34], the random forest [35], and the logistic regression [36], evaluated on the Moons dataset and the Circles dataset, respectively. The techniques are ranked for each classifier in descending order of F_1 score, with Rank 1 representing the best-performing method.

For the Moons dataset, the results presented in Table VI indicate that NEXUT-PlusMinus achieves the highest F_1 score rank for the k-NN classifier, the multilayer perceptron, and the random forest. NEXUT-All obtains the highest rank for the logistic regression. For the decision tree classifier, NCL and NB-Tomek share the highest F_1 score rank. NCL also secures the highest rank for the naive Bayes classifier. Considering F_1 scores across all techniques, NEXUT-PlusMinus demonstrates the highest overall average rank.

For the Circles dataset, the results presented in Table VII indicate that NEXUT-All achieves the highest F_1 score rank for the decision tree classifier and the random forest. NCL secures the highest rank for the k-NN classifier. For the multilayer perceptron, the naive Bayes, and the logistic regression, several techniques demonstrate the highest F_1 score rank, notably NEXUT-All, NEXUT-Plus, NEXUT-PlusMinus, AllKNN, NCL, OSS, and the original dataset. It is important to note that, for these specific classifiers, all techniques except NB-Tomek exhibit comparable performance. Considering the overall F_1 scores for all techniques on the Circles dataset, NEXUT-All achieves the highest average rank.

F. Discussion

The experimental results demonstrate that the proposed NEXUT variants consistently improve classification performance across multiple classifiers and datasets. Specifically, NEXUT-All yields superior F_1 scores when applied to four classifiers: the decision tree, the multilayer perceptron (MLP), the random forest, and the logistic regression. These results indicate the consistent performance of NEXUT-All across different learning models. NEXUT-Plus performs best with the decision tree, the k-NN, the MLP, and the random forest classifiers, while NEXUT-PlusMinus excels with the random forest and the logistic regression classifiers. These findings highlight the effectiveness of the NEXUT approach in selectively removing extreme anomalous negative instances while preserving the overall class distribution.

When compared with established undersampling techniques such as AllKNN [24], NCL [25], OSS [23], and NB-Tomek [27], NEXUT achieves consistently higher or comparable performance across multiple classifiers. As shown in Table V, NEXUT-All significantly outperforms OSS and NB-Tomek on the logistic regression and the random forest classifiers at the 95% confidence level. This result aligns with previous findings showing that OSS [23] and NB-Tomek [27] sometimes removed informative instances, thereby reducing classifier performance. Similarly, NCL [25], which was designed to clean overlapping examples, was sometimes too aggressive in certain domains.

These findings are consistent with Batista et al. [42], who observed that Tomek-based approaches were effective at reducing class overlap. However, they sometimes removed informative instances, which degraded classifier performance. In addition, Drummond and Holte [43] reported that random or overly aggressive undersampling could eliminate useful majority instances. Chawla et al. [44] emphasized the importance of maintaining the class distribution to achieve balanced decision boundaries. More recently, Blagus and Lusa [45] noted that oversampling methods, such as SMOTE and its variants, could alter data variability and introduce correlations between samples, which reduced classifier robustness.

By contrast, NEXUT minimizes these drawbacks by targeting only extreme anomalous negatives. It also avoids excessive data removal and synthetic data generation, thereby preserving meaningful sample diversity while enhancing classification performance.

Despite these advantages, NEXUT has some limitations. It is primarily designed for binary class-imbalanced datasets, and its extension to multi-class scenarios requires further investigation. In cases of extreme data sparsity within the majority class, NEXUT's ability to retain representative instances is reduced. This limitation may affect classifiers that rely heavily on boundary information. While NEXUT is computationally efficient for moderate-sized datasets, further optimization will be necessary for large-scale applications.

In terms of practical applications, NEXUT is well-suited to domains where class imbalance is prevalent and accurate minority detection is critical. For example, in medical diagnosis, detecting rare diseases requires minimizing false negatives to avoid life-threatening errors. In fraud detection, reducing the influence of anomalous majority instances improves the identification of rare fraudulent transactions. In industrial anomaly detection, NEXUT enhances reliability by retaining representative operational data while isolating rare faults. Similarly, in cybersecurity, malicious activities often represent only a very small fraction of network traffic. In such cases, NEXUT helps to preserve decision boundaries that are critical for intrusion detection. By reducing the impact of extreme anomalous negatives, while maintaining meaningful class distributions, NEXUT improves classifier reliability in real-world scenarios where parameter tuning is challenging.

Overall, NEXUT demonstrates higher predictive performance compared to conventional undersampling methods while preserving the class distribution. Future work should explore integrating NEXUT with oversampling techniques to further enhance performance and address more challenging imbalanced datasets.

V. Conclusion

In this research, the Negative Extreme Anomalous Undersampling Technique (NEXUT) is proposed to address the imbalanced classification problem. Unlike other undersampling techniques from the imbalanced-learn Python package, which require user-defined parameters, NEXUT is parameter-free. It outperforms the existing undersampling techniques on continuous, binary, class-imbalanced datasets across the six classifiers. NEXUT applies negative extreme anomalous scores to identify negative instances for elimination. The negative instances selected by NEXUT may be noisy, affecting both the minority and majority classes, or they may simply be redundant. NEXUT effectively addresses overlapping instances between classes while maintaining the data distribution of each class. Moreover, it reduces the size of the majority class, resulting in a more balanced dataset. Thus, the proposed technique helps rebalance class instances.

For future work, NEXUT will be improved by integrating an oversampling technique to enhance performance efficiency. It is suggested to prioritize oversampling the minority class to balance the data, followed by applying undersampling to the majority class to remove noise and potentially overlapping instances.

Acknowledgment

We gratefully acknowledge the financial support provided by the Science Achievement Scholarship of Thailand (SAST) and the Applied Mathematics and Computational Science (AMCS) program. We also thank the Department of Mathematics and Computer Science, Faculty of Science, Chulalongkorn University, Thailand, for their support. Our sincere appreciation also goes to the anonymous reviewers for their constructive comments, which greatly enhanced this manuscript. The authors confirm that a large language model (LLM) was employed to improve the linguistic quality and readability of this manuscript. However, the integrity of the scientific content and the validity of the conclusions presented remain solely with the authors.

TABLE III. THE F_1 SCORE PERFORMANCE ON THE DECISION TREE CLASSIFIER (C4.5), THE K-NEAREST NEIGHBOR CLASSIFIER ($k = 3$), AND THE MULTILAYER PERCEPTRON FOR UCI DATASETS

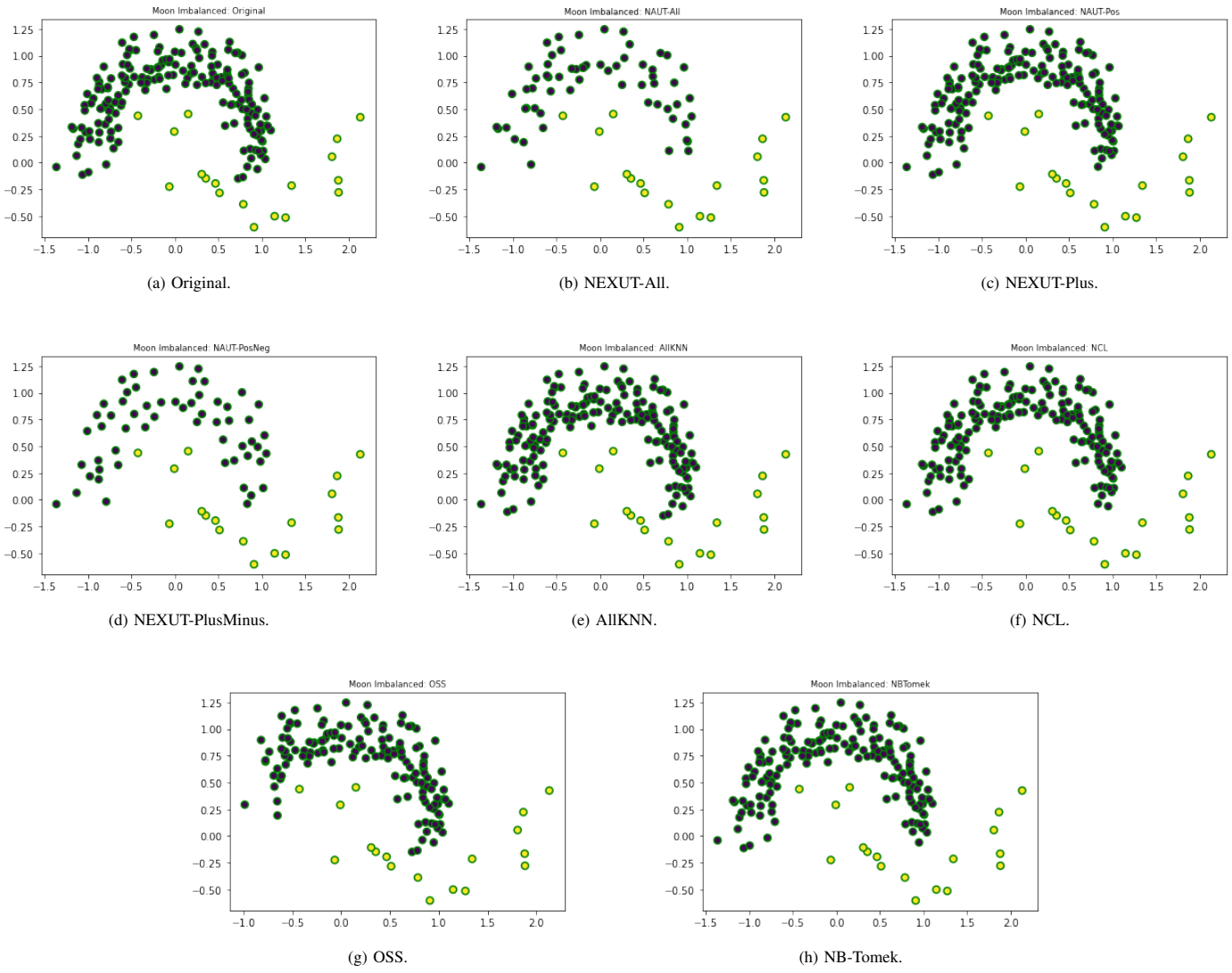
Dataset	Original		NEXUT-All		NEXUT-Plus		NEXUT-PlusMinus		AllKNN		NCL		OSS		NB-Tomek	
	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk
Decision tree classifier (C4.5)																
BreastCancer	0.9389	7	0.9536	4	0.9551	3	0.9552	2	0.9505	5	0.9472	6	0.9259	8	**0.9558**	**1**
Pima	0.6751	6	0.6927	3	**0.6994**	**1**	0.6933	2	0.6718	7	0.6754	5	0.6883	4	0.5319	8
BreastTissue	0.7643	3	**0.7774**	**1**	0.7544	4	0.7089	6	0.6540	8	0.7245	5	0.7656	2	0.6850	7
Haber	0.5447	6	**0.5831**	**1**	0.5642	2	0.5521	3	0.5306	7	0.5461	5	0.5490	4	0.4039	8
Glass	0.9072	7	0.9165	5	0.9289	3	0.9290	2	0.9094	6	0.9251	4	0.8953	8	**0.9382**	**1**
Vehicle	**0.9066**	**1**	0.8954	4	0.8950	5	0.8834	7	0.8916	6	0.8985	2	0.8957	3	0.8079	8
Ecoli	0.8286	7	0.8426	5	0.8472	4	0.8503	3	0.8393	6	0.8508	2	**0.8543**	**1**	0.8025	8
Yeast	0.7787	6	0.7894	3	**0.7962**	**1**	0.7608	7	0.7835	5	0.7895	2	0.7846	4	0.6875	8
Ozone8hr	0.6308	7	0.6310	6	0.6359	3	**0.6464**	**1**	0.6311	5	0.6404	2	0.6315	4	0.5943	8
Libras	0.7461	4	**0.7900**	**1**	0.7252	8	0.7815	2	0.7330	6	0.7257	7	0.7467	3	0.7354	5
Abalone	0.6377	3	0.6157	7	**0.6412**	**1**	0.6235	6	0.6375	4	0.6382	2	0.6262	5	0.6061	8
Ozone1hr	0.5909	7	0.5904	8	0.5977	3	0.5916	6	**0.6065**	**1**	0.5963	4	0.5956	5	0.5992	2
Average rank		5.33		4.00		**3.17**		3.92		5.50		3.83		4.25		6.00
K-nearest neighbor classifier ($K = 3$)																
BreastCancer	0.9664	8	0.9756	3	**0.9765**	**1.5**	**0.9765**	**1.5**	0.9739	5	0.9756	4	0.9681	7	0.9719	6
Pima	0.6586	4	0.6625	3	0.6636	2	0.6505	5	0.6442	6	0.6436	7	**0.6669**	**1**	0.4713	8
BreastTissue	**0.7300**	**1**	0.6692	5	0.6898	3	0.6695	4	0.5481	7	0.6408	6	0.7270	2	0.5249	8
Haber	0.5705	6	**0.6165**	**1**	0.6034	2	0.5884	4	0.5699	7	0.5972	3	0.5748	5	0.4027	8
Glass	0.9151	6	0.9146	7	0.9364	2	**0.9371**	**1**	0.9087	8	0.9358	3	0.9159	5	0.9232	4
Vehicle	**0.9266**	**1**	0.8788	6	0.8880	4	0.8657	7	0.8854	5	0.8954	3	0.9245	2	0.7983	8
Ecoli	0.8622	3	0.8564	6	**0.8700**	**1**	0.8592	5	0.8637	2	0.8534	7	0.8613	4	0.8448	8
Yeast	0.8038	4	**0.8126**	**1**	0.8124	2	0.7774	7	0.8019	6	0.8025	5	0.8086	3	0.6999	8
Ozone8hr	0.5706	7	0.5820	4	0.5792	5	0.5932	2	**0.5995**	**1**	0.5878	3	0.5771	6	0.5442	8
Libras	**0.8861**	**1**	0.8263	7	0.8221	8	0.8359	5	0.8574	3	0.8554	4	0.8830	2	0.8317	6
Abalone	0.6468	8	0.7029	3	0.6986	4	0.7077	2	0.6671	6	**0.7137**	**1**	0.6592	7	0.6901	5
Ozone1hr	0.5028	6	0.5134	3	0.5012	8	0.5160	2	0.5084	5	0.5111	4	0.5021	7	**0.5285**	**1**
Average rank		4.58		4.08		**3.54**		3.79		5.08		4.17		4.25		6.50
Multilayer perceptron																
BreastCancer	0.9616	7	0.9649	4	0.9648	5	0.9651	2	0.9647	6	**0.9659**	**1**	0.8171	8	0.9650	3
Pima	0.6465	4	0.6744	2	**0.6754**	**1**	0.6449	5	0.6163	7	0.6354	6	0.6499	3	0.3953	8
BreastTissue	0.5247	4	0.5258	3	**0.5442**	**1**	0.5164	6	0.4990	7	0.5236	5	0.5394	2	0.4773	8
Haber	0.5850	7	0.5934	5	**0.6383**	**1**	0.5887	6	0.6001	4	0.6154	2	0.6133	3	0.2305	8
Glass	0.6579	8	0.7057	3	0.6808	6	0.7139	2	0.6829	5	0.6875	4	**0.7188**	**1**	0.6737	7
Vehicle	0.8928	2	**0.8948**	**1**	0.8840	5	0.8711	7	0.8845	4	0.8849	3	0.8759	6	0.8423	8
Ecoli	0.8203	2	0.8111	7	0.8021	8	0.8131	5	**0.8284**	**1**	0.8154	4	0.8117	6	0.8187	3
Yeast	0.7829	7	**0.7972**	**1**	0.7933	4	0.7832	6	0.7965	2	0.7950	3	0.7903	5	0.7378	8
Ozone8hr	0.5187	5	0.5187	4	0.5138	7	0.5288	3	0.5170	6	0.5310	2	0.5053	8	**0.5565**	**1**
Libras	0.8542	6	0.7550	8	0.8584	4	0.7663	7	0.8601	3	0.8603	2	**0.8658**	**1**	0.8570	5
Abalone	0.5498	4	0.5831	2	0.5510	3	0.5427	7	0.5491	5	0.5473	6	0.5305	8	**0.6045**	**1**
Ozone1hr	0.4913	8	0.4919	5	0.4924	4	0.4945	2	0.4918	6	0.4925	3	0.4913	7	**0.5048**	**1**
Average rank		5.33		3.75		4.08		4.83		4.67		**3.42**		4.83		5.08

TABLE IV. THE F_1 SCORE PERFORMANCE ON THE NAIVE BAYES, THE RANDOM FOREST, AND THE LOGISTIC REGRESSION FOR UCI DATASETS

Dataset	Original		NEXUT-All		NEXUT-Plus		NEXUT-PlusMinus		AllKNN		NCL		OSS		NB-Tomek	
	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk
Naive Bayes																
BreastCancer	**0.9589**	**1**	0.9536	5	0.9536	4	0.9539	3	0.9498	7	0.9512	6	0.9560	2	0.9386	8
Pima	0.7237	2	0.7186	5	0.7229	3	**0.7266**	**1**	0.6810	7	0.6990	6	0.7187	4	0.5968	8
BreastTissue	0.6370	3	**0.6415**	**1**	0.6348	4	0.6335	5	0.6277	8	0.6322	6	0.6403	2	0.6279	7
Haber	0.5604	7	0.6372	2	0.6207	5	0.6224	4	**0.6523**	**1**	0.6350	3	0.5802	6	0.4018	8
Glass	0.8652	8	**0.8847**	**1**	0.8730	5	0.8766	2	0.8751	3	0.8728	6	0.8716	7	0.8738	4
Vehicle	0.6354	4	0.6321	8	0.6345	6	**0.6411**	**1**	0.6333	7	0.6347	5	0.6384	2	0.6366	3
Ecoli	0.6648	6	0.6409	7	0.6687	3	0.6263	8	**0.6704**	**1**	0.6698	2	0.6676	4	0.6676	5
Yeast	0.6604	4	0.4965	7	0.6662	3	**0.7242**	**1**	0.4968	5	0.4954	8	0.6802	2	0.4968	6
Ozone8hr	**0.5256**	**1**	0.5250	3	0.5214	4	0.5050	7	0.5161	6	0.5161	5	0.5252	2	0.4857	8
Libras	**0.8292**	**1**	0.8152	3	0.8137	4	0.8029	6	0.8178	2	0.8108	5	0.7926	8	0.7973	7
Abalone	0.5890	2	0.5326	8	0.5814	3	0.5775	5	0.5793	4	0.5724	6	**0.5927**	**1**	0.5375	7
Ozone1hr	0.4904	2	0.4864	4	0.4887	3	0.4772	7	0.4852	6	0.4857	5	**0.4909**	**1**	0.4651	8
Average rank		**3.42**		4.50		3.92		4.17		4.75		5.25		**3.42**		6.58
Random forest																
BreastCancer	0.9710	7	0.9742	2	**0.9745**	**1**	0.9742	3	0.9740	4	0.9739	5	0.9709	8	0.9730	6
Pima	0.7181	5	0.7338	2	**0.7341**	**1**	0.7259	4	0.6830	7	0.7005	6	0.7322	3	0.4857	8
BreastTissue	0.7758	3	0.7737	4	0.7931	2	0.7322	5	0.6403	8	0.7071	6	**0.8013**	**1**	0.6639	7
Haber	0.5583	7	**0.6097**	**1**	0.6044	2	0.5980	3	0.5743	5	0.5852	4	0.5727	6	0.3328	8
Glass	0.9326	5	0.9275	6	0.9394	2	0.9386	3	0.9175	8	**0.9400**	**1**	0.9266	7	0.9381	4
Vehicle	**0.9568**	**1**	0.9127	5	0.9225	4	0.9034	6	0.8988	7	0.9234	3	0.9397	2	0.8353	8
Ecoli	0.8502	7	0.8527	5	**0.8673**	**1**	0.8532	3	0.8531	4	0.8511	6	0.8571	2	0.8428	8
Yeast	0.8274	6	0.8297	5	**0.8367**	**1**	0.8260	7	0.8318	4	0.8331	3	0.8353	2	0.7453	8
Ozone8hr	0.5334	8	0.6475	2	0.5704	6	**0.6952**	**1**	0.6357	3	0.6276	5	0.5482	7	0.6314	4
Libras	0.6053	8	**0.7632**	**1**	0.6325	6	0.7469	2	0.6151	7	0.6333	5	0.6404	4	0.6494	3
Abalone	0.5662	8	**0.6563**	**1**	0.5843	6	0.6256	3	0.5941	5	0.6249	4	0.5717	7	0.6469	2
Ozone1hr	0.4921	4	0.5137	3	0.4921	6	0.5230	2	0.4920	8	0.4921	7	0.4921	5	**0.6448**	**1**
Average rank		5.75		**3.08**		3.17		3.50		5.83		4.58		4.50		5.58
Logistic regression																
BreastCancer	0.9638	8	0.9678	6	0.9682	3.5	0.9682	3.5	**0.9695**	**1**	0.9680	5	0.9672	7	0.9691	2
Pima	0.7334	3	**0.7420**	**1**	0.7369	2	0.7315	4	0.6925	7	0.7041	6	0.7282	5	0.5206	8
BreastTissue	0.7853	3	0.7821	5	0.7949	2	0.7823	4	0.6675	7	0.7721	6	**0.7984**	**1**	0.6499	8
Haber	0.5363	7	**0.6556**	**1**	0.6237	5	0.6299	4	0.6517	3	0.6527	2	0.5728	6	0.2636	8
Glass	0.8832	8	0.8892	4	0.8974	3	**0.9043**	**1**	0.8862	7	0.8999	2	0.8879	5	0.8875	6
Vehicle	0.9547	2	0.9433	5	0.9469	4	0.9327	7	0.9376	6	0.9522	3	**0.9560**	**1**	0.8852	8
Ecoli	0.7944	8	0.8276	3	0.8003	7	**0.8358**	**1**	0.8221	4	0.8168	5	0.8125	6	0.8311	2
Yeast	0.6226	8	0.7767	2	0.7162	6	**0.7848**	**1**	0.7492	5	0.7509	4	0.6602	7	0.7510	3
Ozone8hr	0.6195	7	0.6781	2	0.6528	5	**0.6936**	**1**	0.6708	4	0.6778	3	0.6333	6	0.6074	8
Libras	0.7045	8	**0.7766**	**1**	0.7139	7	0.7740	2	0.7144	6	0.7177	4	0.7171	5	0.7247	3
Abalone	0.5047	8	0.5733	2	0.5074	6	0.5223	3	0.5132	5	0.5198	4	0.5053	7	**0.5734**	**1**
Ozone1hr	0.5434	7	0.5710	3	0.5467	6	0.5825	2	0.5545	4	0.5485	5	0.5426	8	**0.6035**	**1**
Average rank		6.42		2.92		4.71		**2.79**		4.92		4.08		5.33		4.83

TABLE V. THE F_1 SCORE STATISTICAL RESULTS FOR UCI DATASETS

F_1 score of	Decision tree				K-NN classifier				MLP				Naive Bayes				Random forest				Logistic regression			
NEXUT-All vs	**Rk+**	**Rk-**	**TS**	**Sign**	**Rk+**	**Rk-**	**TS**	**Sign**	**Rk+**	**Rk-**	**T**	**Sign**	**Rk+**	**Rk-**	**TS**	**Sign**	**Rk+**	**Rk-**	**TS**	**Sign**	**Rk+**	**Rk-**	**TS**	**Sign**
Original	61	17	17	(+)	42	36	36	(+)	59	19	19	(+)	23	55	23	(-)	64	14	14	(+)	72	6	6	(++)
AllKNN	61	17	17	(+)	53	25	25	(+)	54	24	24	(+)	39	39	39	(+/-)	73	5	5	(++)	77	1	1	(++)
NCL	45	33	33	(+)	39	39	39	(+)	42	36	36	(+)	50	28	28	(+)	66	12	12	(++)	67	11	11	(++)
OSS	57	21	21	(+)	37	41	37	(-)	47	31	31	(+)	28	50	28	(-)	54	24	24	(+)	69	9	9	(++)
NB-Tomek	71	7	7	(++)	67	11	11	(++)	52	26	26	(+)	63	15	15	(+)	64	14	14	(+)	65	13	13	(++)
NEXUT-Plus vs																								
Original	59	19	19	(+)	46	32	32	(+)	59	19	19	(+)	35	43	35	(-)	67	11	11	(++)	72	6	6	(++)
AllKNN	66	12	12	(++)	56	22	22	(+)	42	36	36	(+)	56	22	22	(+)	63	15	15	(+)	34	44	34	(-)
NCL	61	17	17	(+)	43	35	35	(+)	39	39	39	(+)	64	14	14	(+)	46	32	32	(+)	20	58	20	(-)
OSS	57	21	21	(+)	43	35	35	(+)	56	22	22	(+)	33	45	33	(-)	56	22	22	(+)	55	23	23	(+)
NB-Tomek	68	10	10	(++)	69	9	9	(++)	53	25	25	(+)	74	4	4	(++)	54	24	24	(+)	49	29	29	(+)
NEXUT-PlusMinus vs																								
Original	47	31	31	(+)	37	41	37	(-)	33	45	33	(-)	33	45	33	(-)	60	18	18	(+)	69	9	9	(++)
AllKNN	57	21	21	(+)	48	30	30	(+)	38	40	38	(-)	36	42	36	(-)	74	4	4	(++)	69	9	9	(++)
NCL	39	39	39	(+/-)	33	45	33	(-)	19	59	19	(-)	37	41	37	(-)	63	15	15	(+)	63	15	15	(+)
OSS	44	34	34	(+)	31	47	31	(-)	31	47	31	(-)	39	39	39	(+/-)	51	27	27	(+)	69	9	9	(++)
NB-Tomek	72	6	6	(++)	75	3	3	(++)	50	28	28	(+)	69	9	9	(++)	64	14	14	(+)	65	13	13	(++)

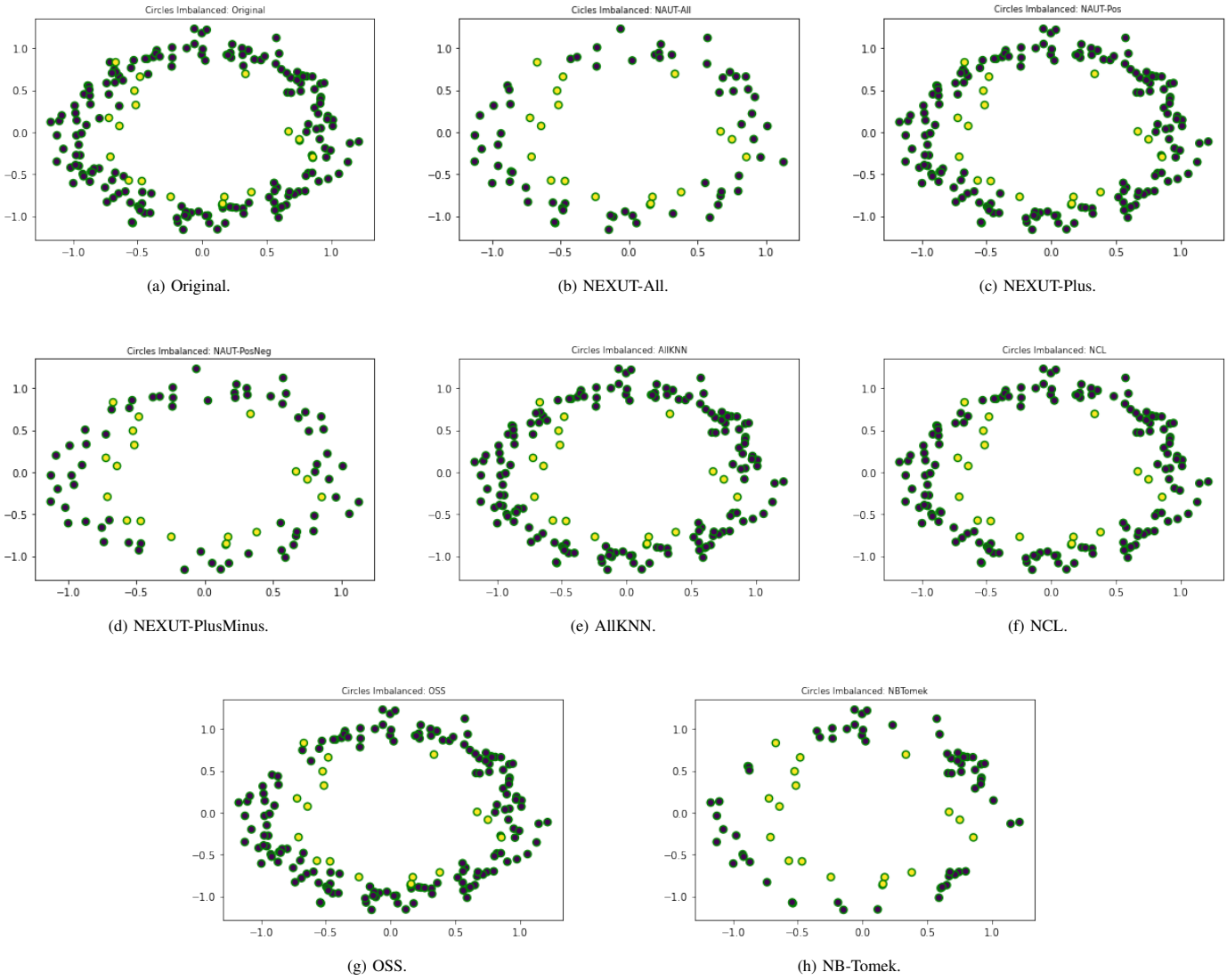


(a) Original. (b) NEXUT-All. (c) NEXUT-Plus.

(d) NEXUT-PlusMinus. (e) AllKNN. (f) NCL.

(g) OSS. (h) NB-Tomek.

Fig. 4. The Moons dataset after using each undersampling technique and without technique.



(a) Original. (b) NEXUT-All. (c) NEXUT-Plus.

(d) NEXUT-PlusMinus. (e) AllKNN. (f) NCL.

(g) OSS. (h) NB-Tomek.

Fig. 5. The Circles dataset after using each undersampling technique and without technique.

TABLE VI. THE F_1 SCORE PERFORMANCES ON THE SIX CLASSIFIERS FOR THE MOONS DATASET

Classifiers	Original		NEXUT-All		NEXUT-Plus		NEXUT-PlusMinus		AllKNN		NCL		OSS		NB-Tomek	
	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk
Decision Tree	0.8366	8	0.8507	6	0.8688	3	0.8652	4	0.8383	7	**0.8692**	**1.5**	0.8643	5	**0.8692**	**1.5**
K-NN classifier	0.9442	8	0.9476	5	0.9605	2	**0.9644**	**1**	0.9460	6	0.9592	3	0.9453	7	0.9559	4
MLP	0.7013	7	0.7970	2	0.7210	4	**0.8001**	**1**	0.6978	8	0.7135	5	0.7278	3	0.7082	6
Naive Bayes	0.8295	5	0.7814	8	0.8369	3	0.7973	6	0.8299	4	**0.8420**	**1**	0.7940	7	0.8378	2
Random Forest	0.8684	8	0.8838	2	0.8822	3	**0.8927**	**1**	0.8691	7	0.8756	5	0.8723	6	0.8776	4
Logistic Regression	0.7030	7.5	**0.7993**	**1**	0.7443	3	0.7875	2	0.7030	7.5	0.7280	5	0.7333	4	0.7257	6
Average Rank		7.25		4.00		3.00		**2.50**		6.58		3.42		5.33		3.92

TABLE VII. THE F_1 SCORE PERFORMANCES ON THE SIX CLASSIFIERS FOR THE CIRCLES DATASET

Classifiers	Original		NEXUT-All		NEXUT-Plus		NEXUT-PlusMinus		AllKNN		NCL		OSS		NB-Tomek	
	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk	F_1	Rk
Decision Tree	0.6216	5	**0.6356**	**1**	0.6239	4	0.6240	3	0.6346	2	0.6208	6	0.6085	7	0.5544	8
K-NN classifier	0.5293	8	0.6332	2	0.6037	5	0.6154	4	0.6201	3	**0.6424**	**1**	0.5825	6	0.5434	7
MLP	**0.4762**	**4**	**0.4762**	**4**	**0.4762**	**4**	**0.4762**	**4**	**0.4762**	**4**	**0.4762**	**4**	**0.4762**	**4**	0.4757	8
Naive Bayes	**0.4762**	**4**	**0.4762**	**4**	**0.4762**	**4**	**0.4762**	**4**	**0.4762**	**4**	**0.4762**	**4**	**0.4762**	**4**	0.4684	8
Random Forest	0.5678	8	**0.6416**	**1**	0.6121	5	0.6403	2	0.6309	4	0.6309	3	0.5997	6	0.5944	7
Logistic Regression	**0.4762**	**4**	**0.4762**	**4**	**0.4762**	**4**	**0.4762**	**4**	**0.4762**	**4**	**0.4762**	**4**	**0.4762**	**4**	0.4738	8
Average Rank		5.50		**2.67**		4.33		3.50		3.50		3.67		5.17		7.67

REFERENCES

[1] M. Saini and S. Susan, "Deep transfer with minority data augmentation for imbalanced breast cancer dataset," *Applied Soft Computing*, vol. 97, p. 106759, 2020.

[2] F. Z. El Hlouli, J. Riffi, M. A. Mahraz, A. Yahyaouy, K. El Fazazy, and H. Tairi, "Credit card fraud detection: Addressing imbalanced datasets with a multi-phase approach," *SN Computer Science*, vol. 5, no. 1, p. 173, 2024.

[3] A. Naboureh, H. Ebrahimy, M. Azadbakht, J. Bian, and M. Amani, "Ruesvms: An ensemble method to handle the class imbalance problem in land cover mapping using google earth engine," *Remote Sensing*, vol. 12, no. 21, p. 3484, 2020.

[4] J. Kalajdjieski, E. Zdravevski, R. Corizzo, P. Lameski, S. Kalajdziski, I. M. Pires, N. M. Garcia, and V. Trajkovik, "Air pollution prediction with multi-modal data and deep neural networks," *Remote Sensing*, vol. 12, no. 24, p. 4142, 2020.

[5] H. A. Ahmed, A. Hameed, and N. Z. Bawany, "Network intrusion detection using oversampling technique and machine learning algorithms," *PeerJ Computer Science*, vol. 8, p. e820, 2022.

[6] W. Chen, K. Yang, Z. Yu, Y. Shi, and C. P. Chen, "A survey on imbalanced learning: latest research, applications and future directions," *Artificial Intelligence Review*, vol. 57, no. 6, p. 137, 2024.

[7] Y. Sun, A. K. Wong, and M. S. Kamel, "Classification of imbalanced data: A review," *International journal of pattern recognition and artificial intelligence*, vol. 23, no. 04, pp. 687–719, 2009.

[8] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on knowledge and data engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[9] N. V. Chawla, N. Japkowicz, and A. Kotcz, "Special issue on learning from imbalanced data sets," *ACM SIGKDD explorations newsletter*, vol. 6, no. 1, pp. 1–6, 2004.

[10] P. Branco, L. Torgo, and R. P. Ribeiro, "A survey of predictive modeling on imbalanced domains," *ACM Computing Surveys (CSUR)*, vol. 49, no. 2, pp. 1–50, 2016.

[11] S. Kotsiantis and P. Pintelas, "Mixture of expert agents for handling imbalanced data sets," *Annals of Mathematics, Computing & Teleinformatics*, vol. 1, no. 1, pp. 46–55, 2003.

[12] I. Tomek, "Two modifications of cnn," *IEEE Trans. Systems, Man and Cybernetics*, vol. 6, pp. 769–772, 1976.

[13] G. Lemaître, F. Nogueira, and C. K. Aridas, "Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 559–563, 2017.

[14] I. Mani and I. Zhang, "knn approach to unbalanced data distributions: a case study involving information extraction," in *Proceedings of workshop on learning from imbalanced datasets*, vol. 126, 2003.

[15] C. Chiamanusorn and K. Sinapiromsaran, "Extreme anomalous oversampling technique for class imbalance," in *Proceedings of the 2017 International Conference on Information Technology*. ACM, 2017, pp. 341–345.

[16] S. Rezvani and X. Wang, "A broad review on class imbalance learning techniques," *Applied Soft Computing*, vol. 143, p. 110415, 2023.

[17] S. S. Rawat and A. K. Mishra, "Review of methods for handling class imbalance in classification problems," in *International Conference on Data, Engineering and Applications*. Springer, 2022, pp. 3–14.

[18] N. U. Niaz, K. N. Shahariar, and M. J. Patwary, "Class imbalance problems in machine learning: A review of methods and future challenges," in *Proceedings of the 2nd International Conference on Computing Advancements*, 2022, pp. 485–490.

[19] S. Sharma and A. Gosain, "Addressing class imbalance in remote sensing using deep learning approaches: a systematic literature review," *Evolutionary Intelligence*, vol. 18, no. 1, p. 23, 2025.

[20] D. L. Wilson, "Asymptotic properties of nearest neighbor rules using edited data," *IEEE Transactions on Systems, Man, and Cybernetics*, no. 3, pp. 408–421, 1972.

[21] P. Hart, "The condensed nearest neighbor rule (corresp.)," *IEEE transactions on information theory*, vol. 14, no. 3, pp. 515–516, 1968.

[22] W.-C. Lin, C.-F. Tsai, Y.-H. Hu, and J.-S. Jhang, "Clustering-based undersampling in class-imbalanced data," *Information Sciences*, vol. 409, pp. 17–26, 2017.

[23] M. Kubat, S. Matwin *et al.*, "Addressing the curse of imbalanced training sets: one-sided selection," in *Icml*, vol. 97. Nashville, USA, 1997, pp. 179–186.

[24] I. Tomek, "An experiment with the edited nearest-neighbor rule," *IEEE Transactions on systems, Man, and Cybernetics*, vol. 6, no. 6, pp. 448–452, 1976.

[25] J. Laurikkala, "Improving identification of difficult small classes by balancing class distribution," in *Conference on Artificial Intelligence in Medicine in Europe*. Springer, 2001, pp. 63–66.

[26] C. Bunkhumpornpat, K. Sinapiromsaran, and C. Lursinsap, "Mute: Majority under-sampling technique," in *2011 8th international conference on information, communications & signal processing*. IEEE, 2011, pp. 1–4.

[27] P. Vuttipittayamongkol and E. Elyan, "Neighbourhood-based undersampling approach for handling imbalanced and overlapped data," *Information Sciences*, vol. 509, pp. 47–70, 2020.

[28] C. Bunkhumpornpat, K. Sinapiromsaran, and C. Lursinsap, "Safe level graph for majority under-sampling techniques," *Chiang Mai Journal of Science*, vol. 41, no. 5.2, pp. 1419–1428, 2014.

[29] X. Liang, Y. Gao, and S. Xu, "Ase: Anomaly scoring based ensemble learning for highly imbalanced datasets," *Expert Systems with Applications*, vol. 238, p. 122049, 2024.

[30] C. Shi, X. Li, J. Lv, J. Yin, and I. Mumtaz, "Robust geodesic based outlier detection for class imbalance problem," *Pattern Recognition Letters*, vol. 131, pp. 428–434, 2020.

[31] J. R. Quinlan, *C4. 5: programs for machine learning*. Elsevier, 2014.

[32] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE transactions on information theory*, vol. 13, no. 1, pp. 21–27, 1967.

[33] S. Haykin, *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1994.

[34] I. Rish *et al.*, "An empirical study of the naive bayes classifier," in *IJCAI 2001 workshop on empirical methods in artificial intelligence*, vol. 3, no. 22, 2001, pp. 41–46.

[35] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[36] D. W. Hosmer Jr, S. Lemeshow, and R. X. Sturdivant, *Applied logistic regression*. John Wiley & Sons, 2013, vol. 398.

[37] M. Lichman *et al.*, "Uci machine learning repository," 2013.

[38] I. B. Khediri, C. Weihs, and M. Limam, "Kernel k-means clustering based local support vector domain description fault detection of multimodal processes," *Expert Systems with Applications*, vol. 39, no. 2, pp. 2166–2171, 2012.

[39] M. Bekkar, H. K. Djemaa, and T. A. Alitouche, "Evaluation measures for models assessment over imbalanced data sets," *J Inf Eng Appl*, vol. 3, no. 10, 2013.

[40] L. A. Jeni, J. F. Cohn, and F. De La Torre, "Facing imbalanced data–recommendations for the use of performance metrics," in *2013 Humaine association conference on affective computing and intelligent interaction*. IEEE, 2013, pp. 245–251.

[41] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine learning research*, vol. 7, no. Jan, pp. 1–30, 2006.

[42] G. E. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD explorations newsletter*, vol. 6, no. 1, pp. 20–29, 2004.

[43] C. Drummond, R. C. Holte *et al.*, "C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling," in *Workshop on learning from imbalanced datasets II*, vol. 11, no. 1-8, 2003.

[44] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[45] R. Blagus and L. Lusa, "Smote for high-dimensional class-imbalanced data," *BMC bioinformatics*, vol. 14, no. 1, p. 106, 2013.

Blockchain Enabled Healthcare Supply Chain: Review, Case Study and Future Opportunities

Muhammad Saad[1],Kamran Ali[2], Muhammad Awais Javed[3], Ahmad Naseem Alvi[4], Ahmed Alfakeeh[5]*
Department of Electrical Engineering, COMSATS University Islamabad, 45550, Islamabad, Pakistan[1,2,3,4]
Department of Information Systems, King Abdul Aziz University, Jeddah, Saudi Arabia[5]

***Abstract*—Blockchain is a major component of future smart healthcare that can improve the security, reliability, trust and automation of healthcare supply chain processes. Blockchain has several applications in areas such as medicine procurement, supply chain tracking, and drug traceability. In this study, we present a review of recent works in the area of Blockchain-enabled healthcare supply chains with a particular focus on pharmaceutical supply chains. We categorize the literature into three major areas namely, procurement, asset management, and system efficiency improvement. We also present a case study of efficient smart contracts for the pharmaceutical supply chain in which we identify different stakeholders of the pharmaceutical supply chain and develop different functions and tasks to be performed by each stakeholder and their interactions with each other. We implement the proposed smart contract in Remix Integrated Development Environment (IDE) using solidity language and evaluate the transaction cost of each function used in the smart contract. Lastly, we also present future opportunities on using Blockchain-enabled healthcare supply chain.**

***Keywords*—Blockchain; smart contracts; healthcare; supply chain**

I. Introduction

Blockchain is an upcoming technology that has applications in many domains. Many application areas such as transportation, health care, banking, etc. can benefit from Blockchain technology [1], [2], [3], [4], [5], [6], [7]. The essence of Blockchain is to secure the data in a distributed manner without the need of a central node for authentication purposes. Thus, Blockchain is a concept that can revolutionize the way the data is secured and managed [8], [9], [10], [11], [12].

The key concept of Blockchain is to store data in the form of blocks that are connected such that the current block contains a hash function of the previous block. This connectivity property as well as the acceptability of only a pre-defined hash function makes Blockchain super secure. To change data in one block, it is needed to tamper data in other blocks too [13], [14], [15], [16]. Moreover, generation of the pre-defined hash function is a computationally heavy task that requires nodes with large processing capacity known as mining nodes [17], [18], [19], [20], [21]. Similarly, a block is only accepted in a Blockchain if more than half of the nodes give consensus to the data. This property also provides additional security to the Blockchain. Blockchain is thus an intelligent way of securing data in a distributed manner [22], [23], [24], [25], [26], [27].

Smart contracts are programs that implement certain conditions as per a particular contract. Contrary to traditional contracts, smart contracts can automate the way contracts are managed and smartly use the technology to check application-specific conditions. Smart contracts provide a reliable solution that does not require a trustable central authority. Smart contracts have many applications in scenarios, where application reliability depends on fulfilling certain conditions [28], [29], [30], [31], [32], [33]. As an example, in a medical supply chain, the temperature of the medicines needs to be maintained. Hence, a smart contract can be developed to monitor the temperature of medical supplies. It is important to note that smart contracts require inputs from the Internet of Things (IoT) sensors and the transmission of data to a server. Based on these inputs, smart contracts can check the permissible conditions issue directions, and make appropriate decisions. Smart contracts are managed using Blockchain technology. Thus, Blockchain-enabled smart contracts are a secure way of implementing and automating checks [34], [35], [36], [37], [38].

The focus of this study is to provide a survey of recent work related to Blockchain and smart contracts in the area of healthcare supply chain. The particular focus of this study will be on the pharmaceutical supply chain. A traditional healthcare supply chain requires manual negotiations and interactions between manufacturers and pharmacies. The goal of the smart contracts-based supply chain is to automate the process of medicine procurement. Another challenge that is part of the healthcare supply chain is the manual checking of different medicine-related parameters such as acceptable temperature while transporting, expiry date of medicines, etc. The smart contracts can include different checks to verify these conditions before purchasing of medicines.

The main contribution of our work is summarized as follows:

- We present a review of recent work related to using Blockchain in healthcare environment. We categorize the related works into three major groups. The first is related to procurement, the second is based on asset management, and the third one is related to system efficiency improvement.
- We present a case study in which we propose a framework of a smart contracts-based pharmaceutical supply chain and identify various stakeholders in the process including the Healthcare Management System (HMS), manufacturers, distributors, pharmacies, and hospitals. We develop different functions for each stakeholder that can be embedded in the smart contracts along with various associated conditions. We

*Corresponding authors.

also develop a sequence of events to highlight the interactions between different stakeholders in the smart contracts-based pharmaceutical supply chain. We implement the proposed smart contract in Remix IDE using solidity language and investigate the transaction gas cost of various functions and methods.

- We present several future opportunities related to the utilization and implementation of Blockchain for the healthcare supply chain.

The rest of the study is organized as follows: Section II presents an overview of the healthcare supply chain, related works regarding the use of Blockchain and smart contracts in different systems, and an overview of different survey papers related to Blockchain-enabled healthcare. Section III presents the classification of work in the area of Blockchain-enabled healthcare supply chain. Section IV, Section V and Section VI present a review of recent works related to procurement, asset management, and improving system efficiency in the Blockchain-enabled healthcare supply chain, respectively. Section VII presents a case study in which we present a proposed smart contract along with its various functions and methods, and highlight the sequence diagram of the smart contract. Furthermore, Section VII also describes smart contract implementation on Remix IDE and presents smart contract cost-related results. Section VIII presents different open challenges and future opportunities. Finally, Section IX present the conclusions.

II. Related Work

A. Overview of Healthcare Supply Chain

The healthcare supply chain refers to the interconnection of different procedures, processes, and networks that enable the distribution of medicines and other healthcare items from manufacturers to patients. Our particular focus in this study is on the pharmaceutical supply chain which starts with the research and development of a drug. Different trials and tests are conducted to verify the effectiveness of the drug. After the verification phase, the drug is manufactured on a large scale by different manufacturers that ensure quality assurance by following different quality control procedures [39], [40], [41]. The drugs are purchased by different distributors and further sold to different pharmacies and hospitals. A drug regulation authority controls the safety and quality of the product and ensures that different regulations are met. Finally, the drug is purchased by the end customer [42], [43].

During the supply chain, different processes are involved and interaction between different stakeholders takes place. Monitoring and tracking of these processes are needed to efficiently control them [44], [45], [46]. The related data needs to be efficiently and securely stored. Moreover, the privacy of communication between stakeholders needs to be maintained. All of these issues can be handled by Blockchain and smart contracts, thus improving the efficiency of the supply chain [47], [48].

B. Use of Blockchain for Different Applications

Several works have investigated the use of IoT, Blockchain, and smart contract technology for different applications. In [3], the authors presented a smart contract validation mechanism for supply chain applications. A smart contract was created for better use of data structures in the mining process at the miner nodes.

In [5], a Secure Electronic, Health Medical Record (EHR/EMR) solution was proposed using Blockchain technology. In the proposed solution, data or medical records are shared in a secure manner utilizing various techniques such as multiple authentication schemes for improved security and access control of Medical Reports (MR), data management, and other safety requirements. The proposed technique consists of three phases, first is the authentication by using a Quantum Cryptography Algorithm (QCA), the second one is the encryption by using Advanced Encryption Standard (AES) and the last is the data retrieval by using a Secure Hash Algorithm (SHA).

In [6], the authors investigated the use of the Ethereum Blockchain platform for educational systems. A smart contract for online examination system was created using the Ethereum Blockchain. The system allows to use of the NodeJS server runtime environment and the MongoDB database system. The system utilized several examination related data to develop the smart contract.

In [7], the authors proposed a Mobile-Healthcare algorithm using the Blockchain network. The proposed algorithm develops smart contracts between doctors and patients by using Blockchain. The patients send different sensor data to the nodes participating in the Blockchain. The work minimizes the delay between the patient and the doctor in case of emergency. In a short time, the doctor can send the medical report to the patient in a secure manner. The work also discussed about the 51% attack on the system in which majority mining is done by a particular group and has control over it.

In [8], the authors proposed a relationship between big data analytics, Blockchain, and the Internet of Things (IoT). Blockchain is a secured database technology for real-time data transition and IoT is used to connect different devices, sensors, and databases over the Internet. Blockchain resolves two major problems in IoT applications, one is to develop trust between nodes, the second is to ensure the data privacy.

In [10], the authors proposed the use of Blockchain technology to create an upgraded diabetes Electronic Medical Record (EMR). The medical information was separated into two sections. The data of the medical institution was in one category, and the data of the patient was in another category. The patient's medical records were stored with authentication, privacy, and security such that no physical medical documents were required.

In [11], the authors proposed a solution for secure patient data sharing using the hyperledger fabric framework-based permissioned Blockchain network. Hyperledger fabric was used as a private Blockchain which can be programmed as required. A new member of the Hyperledger fabric must be enrolled through the Membership Service Provider (MSP). A smart contract was written in the hyperledger of fabric with the help of Chaincode.

In [12], the authors proposed the safe and reliable sharing of data between different nodes by using the Blockchain. With

the help of the decentralized database, the work proposed a secure financial data sharing. In [49], the authors discussed different challenges in the supply chain and presented the solution to these problems with Blockchain technology. It was shown that the Blockchain improves the efficiency of supply chain.

In [22], the authors discussed use of Blockchain to conserve perishable food products in the supply chain. Everyone from the farm to the customer can see the temperature of the product with the use of a distributed ledger. The data in the Blockchain can not be changed. As a result, the proposed technology can reduce different types of food losses.

In [23], the authors discussed the agri-food supply chain system with different problems related to trust, lack of transparency, economic losses, and enterprise brand value. It was shown that the food industry supply chain can be improved by using Blockchain.

In [24], the challenges related to the medical supply chain were discussed. The current supply chain processes fail to provide security, transparency, and trust-like features. Also, the system is centralized and cause failure in case of any fault in the central server. By using Blockchain technology, the authors provided secure information exchange between different stakeholders. The InterPlanetary File System (IPFS) storage system was used to store, fetch, and share data in the Blockchain.

The utilization of Blockchain in vehicular networks was discussed in [25]. Data management for applications such as traffic control, route scheduling, and information exchange was enabled using Blockchain. In [26], a review of different works related to security and privacy of Blockchain was presented.

A Blockchain-based smart contract development environment was proposed in [27]. The work consists of two parts, the first part was a novel transaction framework using smart contract and second part was comparison with the current available smart contracts.

In [28], authors proposed a Blockchain based healthcare system. Every hospital uploads the patient's record on the Blockchain and with the help of a mobile application, patients or authorized users can retrieve records on the mobile. The Blockchain in the application ensures data security and data authenticity.

In [29], authors proposed the use of Blockchain's decentralized nature to handle challenges such as frequent network disconnectivity. Integrating IoT with Blockchain was proposed in [30] for reducing the appointment time of doctors. Blockchain technology provides an active feedback loop to improve the tracking of patient health related data.

In [31], authors presented Blockchain-based smart grid cybersecurity techniques. The data was protected by storing it in a Blockchain network and having it validated by peers in the same network. Advanced smart grid activities like load forecasting, local redistribution, and disaster recovery could all benefit from the validated data.

C. Comparison with Other Related Survey Papers

We present a comparison of other survey papers related to Blockchain-enabled pharmaceutical supply chains and health care in Table I. While there are many survey papers available on the utilization of Blockchain in health care, there are a limited number of reviews or survey papers on the topic of pharmaceutical supply chain. The work in [50] is focused on the advantages of Blockchain for pharmaceutical cold chain. The review in the paper is mostly related to asset management for improving the pharmaceutical cold chain.

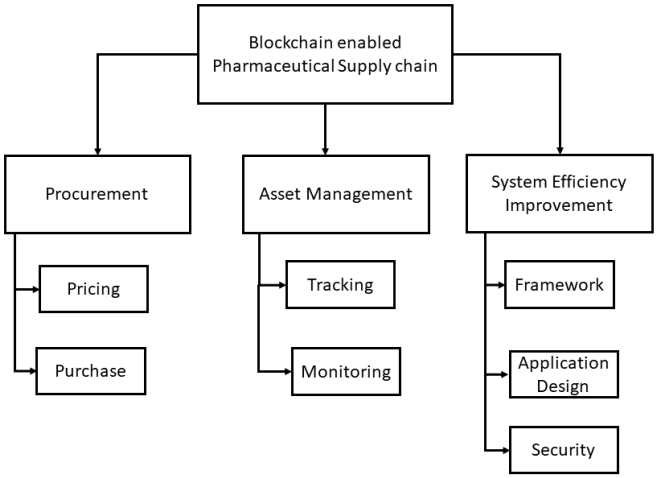


Fig. 1. Classification of work related to Blockchain-enabled pharmaceutical supply chain.

In [51], the review paper is related to handling counterfeit drugs and patient privacy using Blockchain. Similarly, the paper [52] is focused on areas of preventing counterfeit drugs, and tracking and tracing of drugs during their production and distribution life cycle.

The paper in [53] is focused on healthcare applications and presents a survey of work on topics such as data management, healthcare insurance, and supply chain. The survey paper in [54] is focused on the security and interoperability of healthcare applications using Blockchain.

Most of the current survey papers are focused on asset management of pharmaceutical supply chains or other healthcare applications. In addition, most of the review papers are from 2020 to 2022. Most of the recent survey papers are focused on other healthcare applications. Thus, the distinguishing feature of our survey paper is that it reviews three major aspects of the pharmaceutical supply chain and presents recent related works. Moreover, our study also presents a smart contract-based case study to highlight how Blockchain can be utilized for managing the pharmaceutical supply chain.

III. Classification of Work in Blockchain-enabled Pharmaceutical Supply Chain

The work related to the use of Blockchain in the Pharmaceutical supply chain can be classified into three major categories, as shown in Fig. 1. These categories include procurement, asset management, and system efficiency improvement. Further to these categories, the work within these categories can be divided into several subcategories, as discussed below.

TABLE I. COMPARISON OF OUR SURVEY PAPER WITH OTHER RELATED PAPERS

Year	Focus	Procurement	Asset management	Efficiency improvement
2021 [50]	Pharmaceutical cold chain	✗	✓	✗
2022 [51]	Counterfeit drugs Patient privacy	✗	✓	✗
2022 [52]	Counterfeit drugs Tracking Tracing	✗	✓	✗
2020 [53]	Healthcare applications Data management Health insurance Supply chain	✗	✗	✓
2023 [54]	Security Interoperability	✗	✓	✗
Our paper	Pharmaceutical supply chain Smart contract based case study	✓	✓	✓

A. Procurement

Procurement in the supply chain refers to the purchasing of pharmaceutical items and goods. Procurement has a vital role in ensuring that high-quality medicines, vaccines, and raw materials are purchased from the correct sources at a good price.

1) Purchase: The first aspect of procurement is the purchase itself [55], [56]. This means finding the right sources to obtain raw materials. Blockchain can provide a distributed and secure database of these sources [57]. Similarly, purchasing also involves finding reliable manufacturers, distributors, and pharmacies.

2) Pricing: The second aspect of procurement is pricing in which different stakeholders in the supply chain communicate with each other to agree on the pricing of a product. Blockchain can significantly improve the pricing process as it can provide automated smart contracts to evaluate and negotiate the pricing of drugs [58], [59].

B. Asset Management

Asset management is a critical part of the supply chain as it can track and monitor the medicines and drugs during their transportation from manufacturers to distributors and then further to pharmacies. In the context of Blockchain-enabled pharmaceutical supply chain, most of the work can be categorized into tracking and monitoring of medicines [60], [61].

1) Tracking: Tracking involves the management of different medical products during the supply chain. The management includes tracking of raw materials, machinery, and equipment for manufacturing and inventory. With tracking of these products, stakeholders can find the real-time location of medical products and raw materials, and also ensure that the inventory is right quantities. Based on real-time tracking, storage requirements can also be optimized. By using different IoT sensors, Blockchain-based data management, and smart contracts, efficient and secure tracking of inventory can be managed [62], [63].

2) Monitoring: Medical products are dependent on environmental factors to maintain their quality and correct working. Factors such as temperature and humidity need to be maintained within certain thresholds so that these products work at their best. Smart contract-based solutions can regularly monitor these conditions and separate the products that do not follow the required conditions during the supply chain [64].

C. System Efficiency Improvement

Blockchain solutions can be used to improve the supply chain system efficiency such as providing secure data management and prevention against information leakages.

1) Framework: Blockchain-based supply chain framework involves designing solutions that integrate with the current pharmaceutical supply chain system. It also means designing new Blockchain-based systems and process flows to optimize the supply chain. Moreover, the development of IoT solutions, new smart contracts, and data management techniques fall under this category.

2) Application design: The application design involves Blockchain-based new or evolved versions of previous applications related to the pharmaceutical supply chain. Such applications include new payment methods, stakeholder communication, inventory management, and real-time tracking. Such applications enhance the supply chain user experience and improve the efficiency of the supply chain system.

3) Security: Data security is a key challenge for the pharmaceutical supply chain. Data related to medicines is private and sensitive, hence it is critical to find secure solutions [65], [66]. Blockchain provides an efficient data management method that can provide secure access control, immutable data record keeping, privacy of data storage, and resilience to cyber attacks [67], [68], [69].

IV. RECENT WORKS RELATED TO BLOCKCHAIN-BASED PHARMACEUTICAL PROCUREMENT

The recent work related to Blockchain-enabled pharmaceutical procurement is shown in Table II. In [1], a smart

TABLE II. RECENT WORKS RELATED TO BLOCKCHAIN-BASED PHARMACEUTICAL PROCUREMENT

Reference	Blockchain solution	Stakeholders	Procurement process	Results
[1]	Smart contract	GPO Manufacturer Distributor Healthcare provider	Stakeholder registration Admission fees Price update Price negotation Purchase order Delivery status Loyalty discount	Transaction gas cost Execution gas cost
[70]	Blockchain	Competing manufacturers Retailer	Wholesale price Purchase quanity Monetary gains	Blockchain benefit Blockchain cost
[71]	Blockchain Smart contract	Companies Wholsalers Users	Pricing Price history	Application development
[72]	Blockchain Smart contract	Vaccine manufacturer Vaccine unit	Cost sharing contract Revenue sharing contract	Increased profit Supply chain efficiency
[73]	Consortium Blockchain	Patients Manufacturers Health authorities Pharmacy	Value-based pricing	Framework development

contract is proposed that considers multiple pharmaceutical stakeholders such as Group Purchasing Organization (GPO), manufacturer, distributor, and healthcare provider. The proposed work presents detailed functions and methods that are involved in the procurement of medicines. These processes include stakeholder registration, admission fees, updating prices, negotiation of prices, purchase order development, finding the status of delivery, and getting loyalty discounts. The authors implemented the smart contracts in Solidity and evaluated their transaction gas cost and execution gas cost.

The work in [70] developed a Blockchain solution considering two stakeholders, two competing manufacturers and a retailer. For procurement, processes such as wholesale price evaluation, purchase quantity, and monetary gains are considered and related data is stored in a decentralized Blockchain. The work provides an analysis of the cost associated with Blockchain solutions along with their benefits.

In [71], a smart contract-based solution is proposed that considers companies, wholesalers, and users as stakeholders. The proposed work provides pricing methods and also stores pricing history in the Blockchain. This pricing history is stored securely and data changes are not possible. To verify the proposed idea, a mobile application is developed.

The work in [72] developed a Blockchain and smart contract-based technique for the vaccine supply chain. Vaccine manufacturers and vaccine units are the two stakeholders. Two major contracts for procurement include cost sharing and revenue sharing. The proposed technique shows increased profit and improves supply chain efficiency.

A consortium-based Blockchain is proposed in [73]. The work considers four stakeholders namely, patients, manufacturers, health authorities, and pharmacies. The pricing mechanism introduced by the work is based on the value added by each medicine. The paper presents the framework for the proposed value-based pricing mechanism.

V. RECENT WORKS RELATED TO BLOCKCHAIN-BASED PHARMACEUTICAL ASSET MANAGEMENT

The works related to pharmaceutical asset management that used Blockchain is shown in Table III. The work in [74] develops an Ethereum smart contract for asset management in which several features are available such as the quantity of boxes within a lot, pricing of the lot, and drug purchase from trusted manufacturers. A secure decentralized InterPlanetary File System (IPFS) is used as an offchain storage mechanism. Moreover, the system also uses onchain storage as well. Results show the gas cost of the developed smart contract and its security analysis.

In [75], a smart contract is developed for drugs that can be reused and resold. For the reselling application, a verification mechanism from the healthcare authority is proposed. A pricing management system including bidding for the available drugs is developed as part of the smart contract. A reputation ranking system for the resellers is also included. The advantage of this system is that buyers can view the previous selling history and reviews of the reseller. The evaluation of the proposed smart contract presents its gas cost and security analysis.

The work in [76] proposes an IoT and Blockchain integration for asset management. For cost cutting, the minimum distance of the transport route to deliver the medicines is calculated. A mathematical model for route selection from manufacturer to end user is presented. The work also considers environmental change monitoring using IoT sensors. In case any factor is beyond the allowed threshold, a notification is sent to the product owners and customers. Results show reduced latency and response time from the Blockchain network when

TABLE III. RECENT WORKS RELATED TO BLOCKCHAIN-BASED PHARMACEUTICAL ASSET MANAGEMENT

Reference	Blockchain solution	Asset management methods	Results
[74]	Ethereum Blockchain Smart contract	Quantity of boxed within a lot Price of the lot Drugs from trusted manufacturer	Gas cost of smart contract Security analysis
[75]	Ethereum Blockchain Smart contract	Reselling of drugs Verification from healthcare authority Pricing management Bidding Reputation ranking of resellers	Gas cost of smart contract Security analysis
[76]	Blockchain	IoT integration Minimum distance of supply chain Mathematical model Environmental change monitoring Notification to owners and customers	Improve product life time Reduce response time
[77]	Blockchain	Forward and reverse supply chain Reduce transportation cost in forward chain Reduce recall time in reverse chain Faulty drugs recall	Reduced latency Improved throughput
[78]	Smart contract	IoT integration All data signed by stakeholders Stored in Blockchain Record of drug production life cycle	Security analysis Communication cost analysis

the proposed scheme is used. In addition, product lifetime is improved as well.

In [77], the concept of forward and reverse supply chains is introduced and implemented on a Blockchain. Authors develop mathematical models to optimize the supply chain. In the forward direction, the cost is reduced in terms of transportation and inventory costs to supply medicines from the manufacturer to the end user. The model considers vehicle type, reliability of the vehicle, and delivery distance to optimize the cost. In the reverse direction, travel time to recall faulty medicines is reduced. The Blockchain implementation is presented and the work shows reduced latency as compared to the other techniques.

The work in [78] presents a smart contract with IoT integration for asset management. Using IoT sensors, the data related to the drug production life cycle is shared and stored in a Blockchain. Furthermore, the data is signed by all the stakeholders so that no change in it can be made. Results show that the proposed smart contract achieves secure data storage. Results related to communication cost analysis are also presented.

VI. RECENT WORKS RELATED TO BLOCKCHAIN-BASED PHARMACEUTICAL SYSTEM EFFICIENCY

The works related to Blockchain-based pharmaceutical system efficiency improvement is shown in Table IV. In [79], a smart contract framework is developed for the vaccine supply chain and life cycle which improves the system's security and reliability. The proposed system ensures secure record keeping and also enables authentication of all stakeholders. Results show that the proposed framework achieves improved throughput and reduced lost number of data blocks.

The work in [80] proposes a hybrid Blockchain-based framework including both public as well as private Blockchains. A mobile application is developed for all stakeholders. The record-keeping of the drug life cycle is maintained and helps in the detection of counterfeit medicine. Furthermore, IoT is used for monitoring medicines and cryptocurrency is used for purchasing medicines. The work also shows an implementation of the proposed framework.

In [81], a permissioned Blockchain-based framework is developed to enhance the security of the pharmaceutical supply chain. The work proposes the use of secure identity for stakeholders. Similarly, drug information storage is also carried out using secure Blockchain. A secure method is proposed for updating and cancellation of transactions. The system is shown to improve throughput and reduce transaction latency.

The work in [82] proposes a consortium Blockchain to manage the pharmaceutical supply chain. A consensus algorithm is proposed for large-scale traceability of the supply chain. Furthermore, the system is divided into nodes and grouping is done to manage the system efficiently. A ranking system is developed to select trustable nodes to manage transactions in each group. The proposed system improves throughput and reduces transaction latency.

In [83], smart contracts are proposed for dynamic access control in the pharmaceutical supply chain. The work allows the management of permissions at the micro level. Two techniques are used in the proposed system. The first one is the Role-based Access Control (RBAC). This technique is combined with the second Attribute-based Access Control (ABAC) mechanism. The proposed access control improves system throughput in terms of transactions per second [83].

TABLE IV. RECENT WORKS RELATED TO BLOCKCHAIN-BASED PHARMACEUTICAL SYSTEM EFFICIENCY IMPROVEMENT

Reference	Blockchain solution	Framework features	Results
[79]	Public Blockchain Smart contract	Security of vaccine supply chain Smart contract for vaccine life cycle Secure vaccine record keeping Authentication of stake holders	Improved throughput Reduced lost blocks
[80]	Hybrid Blockchain	Mobile application for stake holders Counterfeit medicine detection IoT for medicine monitoring Cryptocurrency for purchase	Framework Implementation
[81]	Permissioned Blockchain	Secure identity of stake holders Secure drug information storage Secure transaction update or cancellation	Improved throughput Reduced transaction latency
[82]	Consortium Blockchain	Consensus algorithm for large scale traceability Divide nodes in to groups Ranking system to select trusted management nodes	Improved throughput Reduced transaction latency
[83]	Smart contract	Dynamic access control Manage permissions at micro level Combination of RBAC and ABAC	Improved throughput

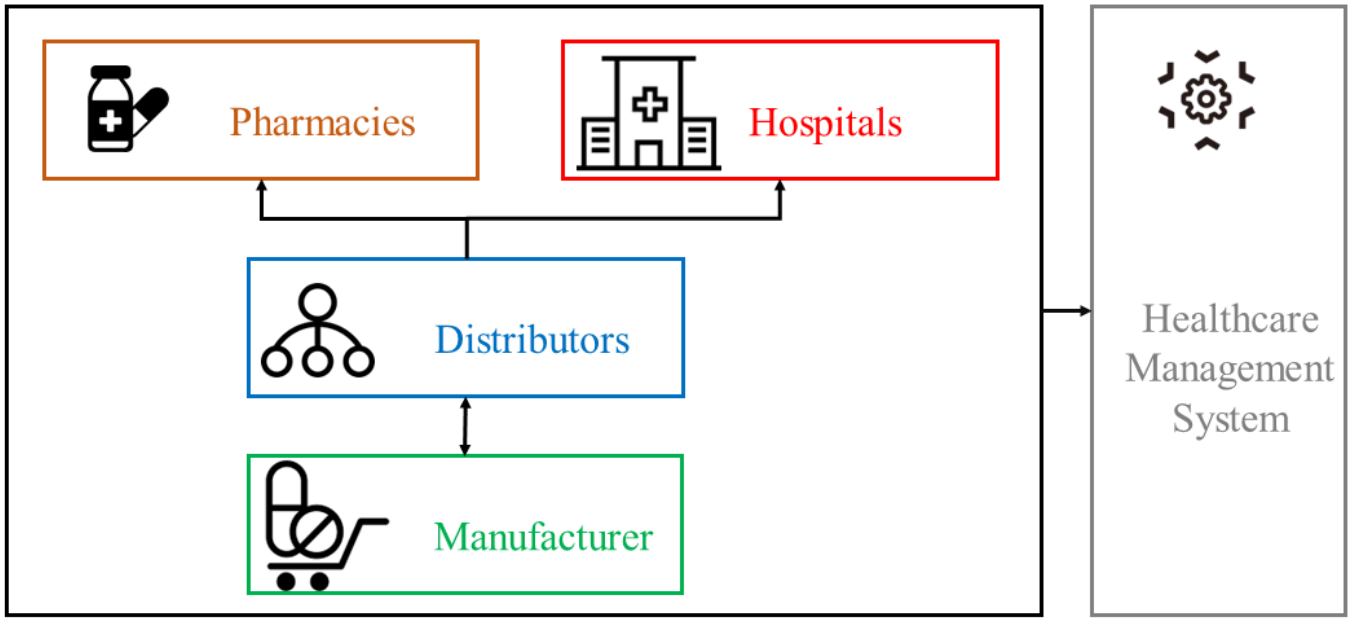


Fig. 2. Components of pharmaceutical supply chain.

VII. CASE STUDY: A PHARMACEUTICAL SMART CONTRACT FRAMEWORK

In this section, we provide a case study related to the pharmaceutical supply chain.

A. Framework

We present a framework for a pharmaceutical supply chain that uses Blockchain-based smart contracts, as shown in Fig. 2. The proposed system has five stakeholders: a Healthcare Management System (HMS), Manufacturers, Distributors, Pharmacies, and Hospitals, all of whom are part of the same decentralized Ethereum-based network. In the contracting process, we use Blockchain technology to provide transparency and data immutability.

B. Components of Smart Contract

There are five main components of the proposed smart contract, where each component is a stakeholder in the system. The role and functions of each stakeholder are described below:

1) Healthcare management system: A Healthcare Management System (HMS) is the centralized server/database that manages the healthcare records, medical processes, and pharmaceutical smart contracts. HMS interacts with different

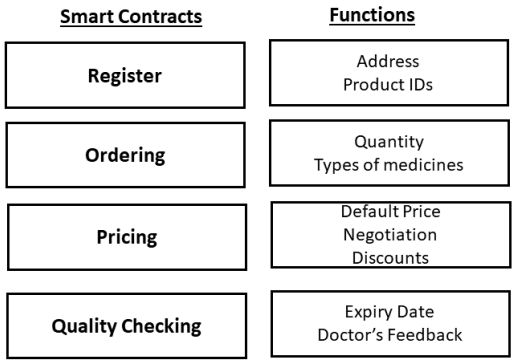


Fig. 3. Different smart contracts and functions for pharmaceutical supply chain.

stakeholders in the pharmaceutical supply chain and manages the large-scale distribution and purchase of medicines. All other stakeholders register themselves in the database of HMS to allow efficient management of the purchase process.

2) Manufacturer: A manufacturer develops and distributes medical and pharmaceutical supplies to the healthcare providers.

3) Distributor: A distributor picks the products from the manufacturing company and sends them to the service providers, i.e., pharmacies and hospitals.

4) Service provider: Pharmaceutical service providers include different pharmacies and hospitals. They purchase medicines from different distributors.

It should be noted that the distributors interact with the manufacturers whereas the service providers communicate with the distributors to enable the supply chain.

C. Description of Smart Contracts

We present the description of different smart contracts managed by the HMS in Fig. 3. There are four major categories of smart contracts, namely registration, ordering, pricing, and quality checking.

The Registration contract is used by all the stakeholders. Each stakeholder registers itself in the central HMS as a manufacturer, distributor, pharmacy, or hospital. The registration is managed using a unique address allocated to each stakeholder. Moreover, each product (i.e., medicine) is also registered and provided a unique ID. Once the registration process is finished, the central database of HMS has a city-wide record of each stakeholder in the supply chain.

The ordering contract is used by the stakeholders to place their orders. Separate ordering contracts are deployed between manufacturers and distributors and between distributors and service providers. The ordering contract has functions to enter the quantity of medicines and type of medicines. Each medicine is categorized into different types for the ease of storing in the database.

The pricing contract enables the purchase of medicines and finalizes the orders at a particular agreed price. It allows manufacturers and distributors to set a default price for medicines. Also, it manages the price negotiation between the stakeholders. Finally, the discounts can be issued on bulk purchases of medicines or any other annual sales event. As in the ordering contract, there are different pricing contracts between each stakeholder.

The last contract is the quality checking contract which is a key feature of the smart contract. This contract can offer many benefits in terms of maintaining the quality of medicines and automating them without the need for manual checking. Conditions such as medicine expiry date can be checked to throw away expired medicines. Different notifications can be issued for medicines that are near the expiry. Similarly, a doctor's feedback on a particular medicine is also incorporated to improve the quality-checking mechanism.

D. Implementation and Cost Analysis

In this section, we present the implementation of the proposed smart contract and its associated cost analysis.

1) Implementation: To implement the proposed smart contract between different stakeholders, we define the functions as discussed in the previous subsection. For this purpose, the Remix Ethereum platform is used. Remix IDE is an open-source platform to write, deploy, and test smart contracts. Remix IDE supports Solidity language, so all the smart contracts are written in the Solidity language. For the Graphic User Interface (GUI) in Remix Ethereum, a Java Virtual Machine (JVM) Environment is available. Ethereum helps in developing decentralized applications and development features such as rapid development, security, and interaction between applications.

We present the implementation of different contracts in the following subsections.

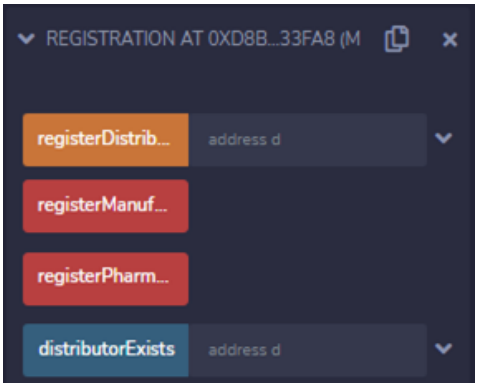


Fig. 4. Registration contract.

2) Registration contract: The Registration contract is shown in Fig. 4. The owner of this contract is HMS which will deploy this contract on its server. This contract deals with the registration process of manufacturers, distributors, pharmacies, and hospitals.

As shown in Fig. 5, this contract has different functions to check issues such as whether the addresses are already used or are available, whether the registration fee is sufficient or not, etc.

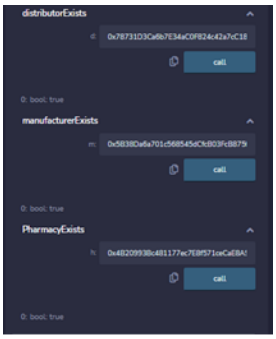


Fig. 5. Check Registered Status

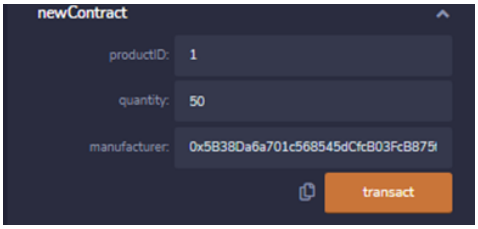


Fig. 6. New contract.

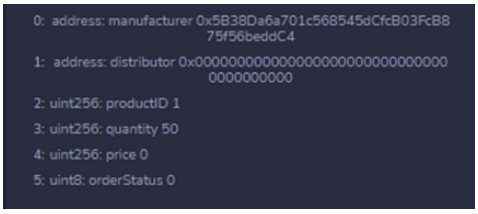


Fig. 7. Contract details.

The built-in keccak256 cryptographic function in Solidity is used for the contracts. Keccak256 converts any number of input arguments to a 32-bit hash. Three arguments (timestamp, contract deployer address, and sender address) are used in this function to generate the contract Address.

We can check the contract details by using the address of the contract, as shown in Fig. 7. The output price and order status, both are updated after the price negotiation, as shown in Fig. 8.

Based on the quantity required, the seller submits a proposed pricing. The potential buyers can show their interest in the item and a mapping with the seller is formed. As an example, the assignDistributor function is used to send orders from the distributor to the service provider, as shown in Fig. 9. This function takes two arguments, contract address and distributor address.

4) Cost analysis: The cost analysis of the smart contract can be conducted by evaluating the gas cost which measures

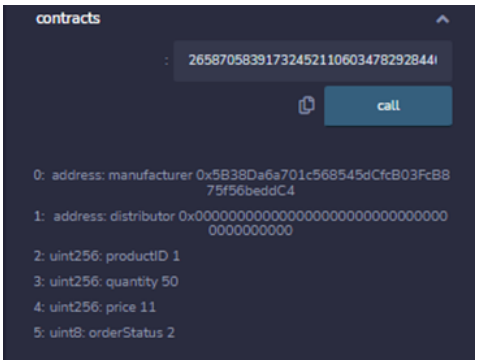


Fig. 8. Updated contract details.

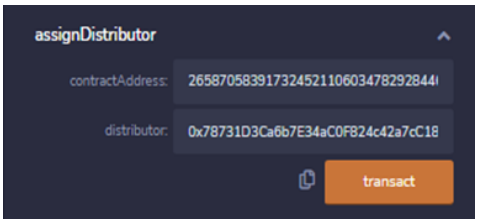


Fig. 9. Assign distributor.

the computational power required for different smart contract functions. Ehereum's native currency ether (ETH) is used for gas payments. The prices of gas are expressed in gwei, which is unit of Ether. One gwei is 10^{-9} ETH. In our case,

- GAS is the unit of computation.
- Gas price is the Ether/Gas (Amount of ether that can be paid per gas)
- Gas limit is the maximum value of gas to be used for a transaction
- Block gas limit is set by the network

The transaction fee is given by the following equation:

Transaction fee = gas used × gas price

Transactions of high gas prices will be processed faster than the low gas prices.

In the Ethereum smart contract, calling different functions require different amount of GAS. GAS is the unit of money in the Ethereum smart contract. GAS depends upon the function complexity and the time. The client can set the upper limit of gas, so for any function, the client does not spend an over amount of gas. So, if the client sets more gas limit, then the transaction will done faster. For measuring the cost, the Ethereum gas station can map the function cost in terms of gas to the corresponding currency amount (such as US dollars).

In Table V, the transaction cost of each function that is used in the registration smart contract is shown. From the above table, the highest transaction cost is for the methods used for the registration of different stakeholders with the HMS. Similarly, registering medicine products with their IDs also incurs a high transaction cost. The transaction cost is reduced for checking the existence of stakeholders in the database.

TABLE V. COST OF REGISTRATION SMART CONTRACT METHODS

Method Name	Transaction Gas Cost
registerManufacturer	49338
registerDistributor	49338
registerPharmacy	49338
registerHospital	49338
ManufacturerExist	24042
DistributorExists	24042
PharmacyExist	24042
HospitalExist	24042
RegisterProduct	39259

TABLE VI. COST OF ORDERING SMART CONTRACT METHODS

Method Name	Transaction Gas Cost
InitiateOrder	58581
SetQuantity	37428
SetMedicineType	43595

In Table VI, the cost of ordering contract methods are shown. There are three methods in this smart contract. The lowest transaction cost in ordering smart contracts is for setting a single order feature. In comparison, initiating a complete order with all the input features renders a higher cost.

TABLE VII. COST OF PRICING SMART CONTRACT METHODS

Method Name	Transaction Gas Cost
SetDefaultPrice	**35621**
RequestNegotiatePrice	**76023**
RespondNegotiatePrice	**45489**
RequestDiscount	**69840**
RespondDiscount	**39284**
assignManufacturer	**58581**
assignDistributor	**58581**
contractStatus	**43595**
newContract	**113207**

In Table VII, the cost of pricing smart contract methods is shown. The first method is the setting of the default price by the entity, that is selling the medicines. The next two methods are used to negotiate prices. The buyer first requests a price that it can offer. Afterwards, the seller either agrees to the buyer's price or sends its final price threshold. Similarly, buyers can request a discount based on membership, etc. The seller responds to it either by agreeing to the discount or sending an updated price it can offer. In addition, there is a method to assign manufacturers to the distributors. Also, there is a method to assign distributors to a pharmacy or a hospital. Moreover, the highest transaction cost is for the newContract method which signs the contract between two stakeholders based on agreement on pricing and quantity. The status of the contract can also be checked using a separate method, as shown in Table VI.

In Table VIII, we show the cost of Quality Checking methods for medicine expiry checks and also for sending doctor's feedback.

TABLE VIII. COST OF QUALITY CHECKING SMART CONTRACT METHODS

Method Name	Transaction Gas Cost
CheckMedicineExpiry	40960
SendDoctorFeedback	57346

It should be noted that we show the gas cost of different methods for different smart contracts. Gas cost (Gwei) depends on the complexity of methods and time. From the results, it can be seen that the gas cost (Gwei) of the newContract method is greater than all other methods of different smart contracts.

VIII. FUTURE OPPORTUNITIES

In this section, we present future opportunities related to the Blockchain-enabled healthcare supply chain.

A. Supply Chain Health Monitoring Metrics

The working range of healthcare supplies in terms of temperature, humidity, and other factors varies from product to product. It is critical to design supply chain health monitoring metrics that consider all of these different parameters. Instead of using various parameters to find thresholds and conditions for smart contracts, it is more effective to use some of these health monitoring metrics. This way smart contracts will be easier to design with less number of functions and methods. This will also reduce the gas cost and transaction cost of the smart contracts.

To develop these health monitoring metrics, mathematical models may be needed that take into account thresholds of parameters with different ranges. A major challenge in this regard is that all these parameters have different operating ranges and units. Hence, it is challenging to design a single metric that gives accurate values of health monitoring. In this regard, multi-criteria decision-making algorithms such as Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) may be utilized.

B. Gas Cost Reduction in Smart Contracts

Smart contracts for supply chains can automate various procedures and processes. However, the major cost associated with it is the gas cost for Blockchain transactions [84], [85]. This cost may increase the overall budget of the supply chain and eventually have an impact on the prices of medicines and healthcare supplies [86], [87].

A major challenge is to reduce the gas cost of smart contracts by using efficient implementation techniques in terms of variables and data structures [88], [89]. The code of smart contracts needs to be optimized for efficient working and reduced transactions. Moreover, the number of methods can be reduced by efficient smart contract design considering the application [90].

C. Interoperability

Blockchain-enabled healthcare solutions need to be inoperable and work with different manufacturers, distributors, pharmacies, and hospitals [91], [92]. Each of the stakeholders has different priorities and preferences regarding the supply chain solutions. Factors such as cost, reliability, and efficiency are considered by each stakeholder, and decisions are made to select a particular solution based on their available budget [93], [94].

It may also happen that different stakeholders may be using different types of smart contracts and Blockchain. The developed smart contracts can be standardized and work for different types of Blockchain and associated environments [95], [96]. Moreover, algorithms and protocols are needed to communicate among Blockchains for transferring of data and tokens [97].

D. Smart Contracts for Healthcare Supplies

While the focus of the current study is on the pharmaceutical supply chain, there are many other types of items and supplies associated with healthcare. For example, procurement and distribution of health-related machine equipment such as ventilators, radiology machines, etc. Similarly, healthcare supplies also involve the distribution of surgical equipment, supplies used in laboratories, and diagnostic items [98], [99], [100].

While pharmaceutical supply chains are more focused on considering environmental factors to maintain the working condition of medicines, other healthcare supplies such as equipment may be more sensitive to road conditions and wear and tear during transportation. Thus, it is challenging to design different smart contracts for different healthcare supply chains considering individual features and characteristics [101], [102].

IX. Conclusion

This study presents a review of Blockchain-enabled healthcare and pharmaceutical supply chains. The study provides an overview of the healthcare supply chain and the uses of Blockchain in different applications. In comparison with the other survey papers on this topic, this study classifies the work done in this area into three major categories. The first is the work related to procurement, the second category is the work related to asset management and the last category is the work related to system efficiency improvement. A survey of recent work in these three areas is presented. The work also presents a case study in which we propose a smart contract involving different stakeholders for the pharmaceutical supply chain. The work describes in detail functions of the proposed smart contract framework. Also, we present details of smart contract implementation and transaction cost analysis. Lastly, we also identify major future research opportunities related to the blockchain-enabled healthcare supply chain.

Acknowledgment

This work was supported by the National Research Program for Universities (NRPU) Project Number: 15363 funded by Higher Education Commission (HEC) Pakistan.

References

[1] I. A. Omar, R. Jayaraman, M. S. Debe, K. Salah, I. Yaqoob, and M. Omar, "Automating procurement contracts in the healthcare supply chain using blockchain smart contracts," *IEEE Access*, vol. 9, pp. 37 397–37 409, 2021.

[2] J. Ahmed, T. N. Nguyen, B. Ali, M. A. Javed, and J. Mirza, "On the physical layer security of federated learning based iomt networks," *IEEE Journal of Biomedical and Health Informatics*, vol. 27, no. 2, pp. 691–697, 2023.

[3] S. Su, K. Wang, and H. S. Kim, "Smartsupply: Smart contract based validation for supply chain blockchain," in *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 2018, pp. 988–993.

[4] T. Gabriel, A. Cornel-Cristian, M. Arhip-Calin, and A. Zamfirescu, "Cloud storage. a comparison between centralized solutions versus decentralized cloud storage solutions using blockchain technology," in *2019 54th International Universities Power Engineering Conference (UPEC)*, 2019, pp. 1–5.

[5] M. Shah, M. Shaikh, V. Mishra, and G. Tuscano, "Decentralized cloud storage using blockchain," in *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, 2020, pp. 384–389.

[6] M. S. Christo, A. M. A., P. S. G., P. C., and R. K. M., "An efficient data security in medical report using block chain technology," in *2019 International Conference on Communication and Signal Processing (ICCSP)*, 2019, pp. 0606–0610.

[7] A. Jain, A. Kumar Tripathi, N. Chandra, and P. Chinnasamy, "Smart contract enabled online examination system based in blockchain network," in *2021 International Conference on Computer Communication and Informatics (ICCCI)*, 2021, pp. 1–7.

[8] M. Rwibasira and R. Suchithra, "A survey paper on consensus algorithm of mobile-healthcare in blockchain network," in *2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*, 2020, pp. 1–5.

[9] P. Manjunath, M. Prakruthi, and P. Gajkumar Shah, "Iot driven with big data analytics and block chain application scenarios," in *2018 Second International Conference on Green Computing and Internet of Things (ICGCIoT)*, 2018, pp. 569–572.

[10] S. K and C. Priya, "Enabling the efficiency of blockchain technology in tele-healthcare with enhanced emr," in *2020 International Conference on Computer Science, Engineering and Applications (ICCSEA)*, 2020, pp. 1–6.

[11] N. Kumar S. and M. Dakshayini, "Secure sharing of health data using hyperledger fabric based on blockchain technology," in *2020 International Conference on Mainstreaming Block Chain Implementation (ICOMBI)*, 2020, pp. 1–5.

[12] Z. Su, H. Wang, H. Wang, and X. Shi, "A financial data security sharing solution based on blockchain technology and proxy reencryption technology," in *2020 IEEE 3rd International Conference of Safe Production and Informatization (IICSPI)*, 2020, pp. 462–465.

[13] A. Lei, H. Cruickshank, Y. Cao, P. Asuquo, C. P. A. Ogah, and Z. Sun, "Blockchain-based dynamic key management for heterogeneous intelligent transportation systems," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 1832–1843, 2017.

[14] A. O. Mahony and E. Popovici, "A systematic review of blockchain hardware acceleration architectures," in *2019 30th Irish Signals and Systems Conference (ISSC)*, 2019, pp. 1–6.

[15] Y. Sakakibara, K. Nakamura, and H. Matsutani, "An fpga nic based hardware caching for blockchain," in *Proceedings of the 8th International Symposium on Highly Efficient Accelerators and Reconfigurable Technologies*, ser. HEART2017. New York, NY, USA: Association for Computing Machinery, 2017. [Online]. Available: https://doi.org/10.1145/3120895.3120897

[16] Y. SAKAKIBARA, S. MORISHIMA, K. NAKAMURA, and H. MATSUTANI, "A hardware-based caching system on fpga nic for blockchain," *IEICE Transactions on Information and Systems*, vol. E101.D, no. 5, pp. 1350–1360, 2018.

[17] F. N. Tareen, A. N. Alvi, A. A. Malik, M. A. Javed, M. B. Khan, A. K. J. Saudagar, M. Alkhathami, and M. H. Abul Hasanat, "Efficient load balancing for blockchain-based healthcare system in smart cities," *Applied Sciences*, vol. 13, no. 4, p. 2411, 2023.

[18] S. Ravi Kumar and M. Goyal, "Tiscmb: design of a highly efficient blockchain consensus model with trust integrated self-correcting miner selection," *International Journal of Information Technology*, vol. 15, no. 4, pp. 1845–1858, 2023.

[19] J. Gao, B. Li, and Z. Li, "Blockchain storage analysis and optimization of bitcoin miner node," in *Communications, Signal Processing, and Systems: Proceedings of the 2018 CSPS Volume III: Systems 7th*. Springer, 2020, pp. 922–932.

[20] G. Lee, J. Park, W. Saad, and M. Bennis, "Performance analysis of blockchain systems with wireless mobile miners," *IEEE Networking Letters*, vol. 2, no. 3, pp. 111–115, 2020.

[21] M. A. Mohamed, S. Mirjalili, U. Dampage, S. H. Salmen, S. A. Obaid, and A. Annuk, "A cost-efficient-based cooperative allocation of mining devices and renewable resources enhancing blockchain architecture," *Sustainability*, vol. 13, no. 18, p. 10382, 2021.

[22] N. N. Ahamed, T. K. Thivakaran, and P. Karthikeyan, "Perishable food products contains safe in cold supply chain management using blockchain technology," in *2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS)*, vol. 1, 2021, pp. 167–172.

[23] S. Menon and K. Jain, "Blockchain technology for transparency in agri-food supply chain: Use cases, limitations, and future directions," *IEEE Transactions on Engineering Management*, pp. 1–15, 2021.

[24] R. W. Ahmad, K. Salah, R. Jayaraman, I. Yaqoob, M. Omar, and S. Ellahham, "Blockchain-based forward supply chain and waste management for covid-19 medical equipment and supplies," *IEEE Access*, vol. 9, pp. 44 905–44 927, 2021.

[25] T. Alladi, V. Chamola, N. Sahu, V. Venkatesh, A. Goyal, and M. Guizani, "A comprehensive survey on the applications of blockchain for securing vehicular networks," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 2, pp. 1212–1239, 2022.

[26] A. A. Mamun, S. Azam, and C. Gritti, "Blockchain-based electronic health records management: A comprehensive review and future research direction," *IEEE Access*, vol. 10, pp. 5768–5789, 2022.

[27] M. Muneeb, Z. Raza, I. U. Haq, and O. Shafiq, "Smartcon: A blockchain-based framework for smart contracts and transaction management," *IEEE Access*, vol. 10, pp. 23 687–23 699, 2022.

[28] S. Yongjoh, C. So-In, P. Kompunt, P. Muneesawang, and R. I. Morien, "Development of an internet-of-healthcare system using blockchain," *IEEE Access*, vol. 9, pp. 113 017–113 031, 2021.

[29] M. Kassab, J. DeFranco, T. Malas, P. Laplante, G. Destefanis, and V. V. G. Neto, "Exploring research in blockchain for healthcare and a roadmap for the future," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 4, pp. 1835–1852, 2021.

[30] K. N. Griggs, O. Ossipova, C. P. Kohlios, A. N. Baccarini, E. A. Howson, and T. Hayajneh, "Healthcare blockchain system using smart contracts for secure automated remote patient monitoring," *Journal of Medical Systems*, vol. 42, pp. 1–7, 2018.

[31] P. Zhuang, T. Zamir, and H. Liang, "Blockchain for cybersecurity in smart grid: A comprehensive survey," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 1, pp. 3–19, 2021.

[32] A. A. Mamun, S. Azam, and C. Gritti, "Blockchain-based electronic health records management: A comprehensive review and future research direction," *IEEE Access*, vol. 10, pp. 5768–5789, 2022.

[33] V. Y. Kemmoe, W. Stone, J. Kim, D. Kim, and J. Son, "Recent advances in smart contracts: A technical overview and state of the art," *IEEE Access*, vol. 8, pp. 117 782–117 801, 2020.

[34] R. Mars, S. Cheikhrouhou, S. Kallel, and A. Hadj Kacem, "A survey on automation approaches of smart contract generation," *The Journal of Supercomputing*, pp. 1–33, 2023.

[35] K. L. Narayana and K. Sathiyamurthy, "Automation and smart materials in detecting smart contracts vulnerabilities in blockchain using deep learning," *Materials Today: Proceedings*, vol. 81, pp. 653–659, 2023.

[36] H. Taherdoost, "Smart contracts in blockchain technology: A critical review," *Information*, vol. 14, no. 2, p. 117, 2023.

[37] J. Leng, W. Sha, Z. Lin, J. Jing, Q. Liu, and X. Chen, "Blockchained smart contract pyramid-driven multi-agent autonomous process control for resilient individualised manufacturing towards industry 5.0," *International Journal of Production Research*, vol. 61, no. 13, pp. 4302–4321, 2023.

[38] E. E. Ameyaw, D. J. Edwards, B. Kumar, N. Thurairajah, D.-G. Owusu-Manu, and G. D. Oppong, "Critical factors influencing adoption of blockchain-enabled smart contracts in construction projects," *Journal of Construction Engineering and Management*, vol. 149, no. 3, p. 04023003, 2023.

[39] M. S. Golan, L. H. Jernegan, and I. Linkov, "Trends and applications of resilience analytics in supply chain modeling: systematic literature review in the context of the covid-19 pandemic," *Environment Systems and Decisions*, vol. 40, no. 2, pp. 222–243, 2020.

[40] P. Chowdhury, S. K. Paul, S. Kaisar, and M. A. Moktadir, "Covid-19 pandemic related supply chain studies: A systematic review," *Transportation Research Part E: Logistics and Transportation Review*, vol. 148, p. 102271, 2021.

[41] J. Sunny, N. Undralla, and V. M. Pillai, "Supply chain transparency through blockchain-based traceability: An overview with demonstration," *Computers & Industrial Engineering*, vol. 150, p. 106895, 2020.

[42] H. Alzoubi, G. Ahmed, A. Al-Gasaymeh, and B. Kurdi, "Empirical study on sustainable supply chain strategies and its impact on competitive priorities: The mediating role of supply chain collaboration," *Management Science Letters*, vol. 10, no. 3, pp. 703–708, 2020.

[43] M. Pournader, Y. Shi, S. Seuring, and S. L. Koh, "Blockchain applications in supply chains, transport and logistics: a systematic review of the literature," *International Journal of Production Research*, vol. 58, no. 7, pp. 2063–2081, 2020.

[44] S. E. Chang and Y. Chen, "When blockchain meets supply chain: A systematic literature review on current development and potential applications," *Ieee Access*, vol. 8, pp. 62 478–62 494, 2020.

[45] F. Goodarzian, H. Hosseini-Nasab, J. Muñuzuri, and M.-B. Fakhrzad, "A multi-objective pharmaceutical supply chain network based on a robust fuzzy model: A comparison of meta-heuristics," *Applied soft computing*, vol. 92, p. 106331, 2020.

[46] P. Gonczol, P. Katsikouli, L. Herskind, and N. Dragoni, "Blockchain implementations and use cases for supply chains-a survey," *Ieee Access*, vol. 8, pp. 11 856–11 871, 2020.

[47] M. M. Queiroz, R. Telles, and S. H. Bonilla, "Blockchain and supply chain management integration: a systematic review of the literature," *Supply chain management: An international journal*, vol. 25, no. 2, pp. 241–254, 2020.

[48] I. Abu-Elezz, A. Hassan, A. Nazeemudeen, M. Househ, and A. Abd-Alrazaq, "The benefits and threats of blockchain technology in healthcare: A scoping review," *International Journal of Medical Informatics*, vol. 142, p. 104246, 2020.

[49] Y. Raj and S. B, "Study on supply chain management using blockchain technology," in *2021 6th International Conference on Inventive Computation Technologies (ICICT)*, 2021, pp. 1243–1247.

[50] S. M. H. Bamakan, S. G. Moghaddam, and S. D. Manshadi, "Blockchain-enabled pharmaceutical cold chain: Applications, key challenges, and future trends," *Journal of Cleaner Production*, vol. 302, p. 127021, 2021.

[51] A. Ghadge, M. Bourlakis, S. Kamble, and S. Seuring, "Blockchain implementation in pharmaceutical supply chains: A review and conceptual framework," *International Journal of Production Research*, vol. 61, no. 19, pp. 6633–6651, 2023.

[52] N. Zakari, M. Al-Razgan, A. Alsaadi, H. Alshareef, L. Alashaikh, M. Alharbi, R. Alomar, S. Alotaibi *et al.*, "Blockchain technology in the pharmaceutical industry: a systematic review," *PeerJ Computer Science*, vol. 8, p. e840, 2022.

[53] L. Soltanisehat, R. Alizadeh, H. Hao, and K.-K. R. Choo, "Technical, temporal, and spatial research challenges and opportunities in blockchain-based healthcare: A systematic literature review," *IEEE Transactions on Engineering Management*, vol. 70, no. 1, pp. 353–368, 2020.

[54] E. R. D. Villarreal, J. García-Alonso, E. Moguel, and J. A. H. Alegría, "Blockchain for healthcare management systems: A survey on interoperability and security," *IEEE Access*, vol. 11, pp. 5629–5652, 2023.

[55] C. M. Harland, L. Knight, A. S. Patrucco, J. Lynch, J. Telgen, E. Peters, T. Tátrai, and P. Ferk, "Practitioners' learning about healthcare supply chain management in the covid-19 pandemic: a public procurement perspective," *International Journal of Operations & Production Management*, vol. 41, no. 13, pp. 178–189, 2021.

[56] H. Skipworth, E. Delbufalo, and C. Mena, "Logistics and procurement outsourcing in the healthcare sector: A comparative analysis," *European Management Journal*, vol. 38, no. 3, pp. 518–532, 2020.

[57] A. Spieske, M. Gebhardt, M. Kopyto, and H. Birkel, "Improving resilience of the healthcare supply chain in a pandemic: Evidence from europe during the covid-19 crisis," *Journal of Purchasing and Supply Management*, vol. 28, no. 5, p. 100748, 2022.

[58] E. Schneller, Y. Abdulsalam, K. Conway, and J. Eckler, *Strategic management of the healthcare supply chain*. John Wiley & Sons, 2023.

[59] W. Bian, X. Yang, S. Li, X. Yang, and G. Hua, "Advantages of 3pls as healthcare supply chain orchestrators," *Computers & Industrial Engineering*, vol. 161, p. 107628, 2021.

[60] R. Jayaraman, K. Salah, and N. King, "Improving opportunities in healthcare supply chain processes via the internet of things and blockchain technology," in *Research Anthology on Blockchain Technology in Business, Healthcare, Education, and Government*. IGI Global, 2021, pp. 1635–1654.

[61] A. Salehi-Amiri, A. Jabbarzadeh, M. Hajiaghaei-Keshteli, and A. Chaabane, "Utilizing the internet of things (iot) to address uncertain home health care supply chain network," *Expert Systems with Applications*, vol. 208, p. 118239, 2022.

[62] M. Safkhani, S. Rostampour, Y. Bendavid, and N. Bagheri, "Iot in medical & pharmaceutical: Designing lightweight rfid security protocols for ensuring supply chain integrity," *Computer Networks*, vol. 181, p. 107558, 2020.

[63] S. Kumar and A. K. Pundir, "Blockchain—internet of things (iot) enabled pharmaceutical supply chain for covid-19," in *Proceedings of the NA International Conference on Industrial Engineering and Operations Management Detroit, Detroit, MI, USA*, 2020, pp. 10–14.

[64] A. Zahedi, A. Salehi-Amiri, N. R. Smith, and M. Hajiaghaei-Keshteli, "Utilizing iot to design a relief supply chain network for the sars-cov-2 pandemic," *Applied soft computing*, vol. 104, p. 107210, 2021.

[65] M. N. M. Bhutta, A. A. Khwaja, A. Nadeem, H. F. Ahmad, M. K. Khan, M. A. Hanif, H. Song, M. Alshamari, and Y. Cao, "A survey on blockchain technology: Evolution, architecture and security," *Ieee Access*, vol. 9, pp. 61 048–61 073, 2021.

[66] H. Guo and X. Yu, "A survey on blockchain technology and its security," *Blockchain: research and applications*, vol. 3, no. 2, p. 100067, 2022.

[67] J. Leng, M. Zhou, J. L. Zhao, Y. Huang, and Y. Bian, "Blockchain security: A survey of techniques and research directions," *IEEE Transactions on Services Computing*, vol. 15, no. 4, pp. 2490–2510, 2020.

[68] E. Zaghloul, T. Li, M. W. Mutka, and J. Ren, "Bitcoin and blockchain: Security and privacy," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 10 288–10 313, 2020.

[69] P. J. Taylor, T. Dargahi, A. Dehghantanha, R. M. Parizi, and K.-K. R. Choo, "A systematic literature review of blockchain cyber security," *Digital Communications and Networks*, vol. 6, no. 2, pp. 147–156, 2020.

[70] B. Niu, J. Dong, and Y. Liu, "Incentive alignment for blockchain adoption in medicine supply chains," *Transportation Research Part E: Logistics and Transportation Review*, vol. 152, p. 102276, 2021.

[71] M. Yoo and Y. Won, "A study on the transparent price tracing system in supply chain management based on blockchain," *Sustainability*, vol. 10, no. 11, p. 4037, 2018.

[72] R. Liu, C. Tan, and C. Zhao, "Pricing and coordination of vaccine supply chain based on blockchain technology," *Internet Research*, vol. 31, no. 6, pp. 2096–2119, 2021.

[73] A. Bhattacharjee, M. Chaudhary, and S. Ranganathan, "Value-based pricing for specialty drugs: Solving the implementation puzzle using blockchain," in *2022 OPJU International Technology Conference on Emerging Technologies for Sustainable Development (OTCON)*, 2023, pp. 1–6.

[74] A. Musamih, K. Salah, R. Jayaraman, J. Arshad, M. Debe, Y. Al-Hammadi, and S. Ellahham, "A blockchain-based approach for drug traceability in healthcare supply chain," *IEEE Access*, vol. 9, pp. 9728–9743, 2021.

[75] M. Debe, K. Salah, R. Jayaraman, and J. Arshad, "Blockchain-based verifiable tracking of resellable returned drugs," *IEEE Access*, vol. 8, pp. 205 848–205 862, 2020.

[76] S. K. Nanda, S. K. Panda, and M. Dash, "Medical supply chain integrated with blockchain and iot to track the logistics of medical products," *Multimedia Tools and Applications*, pp. 1–23, 2023.

[77] D. Agrawal, S. Minocha, S. Namasudra, and A. H. Gandomi, "A robust drug recall supply chain management system using hyperledger blockchain ecosystem," *Computers in biology and medicine*, vol. 140, p. 105100, 2022.

[78] C.-L. Chen, Y.-Y. Deng, C.-T. Li, S. Zhu, Y.-J. Chiu, and P.-Z. Chen, "An iot-based traceable drug anti-counterfeiting management system," *IEEE Access*, vol. 8, pp. 224 532–224 548, 2020.

[79] M. Rehman, I. T. Javed, K. N. Qureshi, T. Margaria, and G. Jeon, "A cyber secure medical management system by using blockchain," *IEEE Transactions on Computational Social Systems*, vol. 10, no. 4, pp. 2123–2136, 2023.

[80] G. Subramanian, A. S. Thampy, N. V. Ugwuoke, and B. Ramnani, "Crypto pharmacy – digital medicine: A mobile application integrated with hybrid blockchain to tackle the issues in pharma supply chain," *IEEE Open Journal of the Computer Society*, vol. 2, pp. 26–37, 2021.

[81] E. S. Babu, I. Kavati, S. R. Nayak, U. Ghosh, and W. Al Numay, "Secure and transparent pharmaceutical supply chain using permissioned blockchain network," *International Journal of Logistics Research and Applications*, pp. 1–28, 2022.

[82] S. Liu, R. Zhang, C. Liu, and D. Shi, "P-pbft: An improved blockchain algorithm to support large-scale pharmaceutical traceability," *Computers in biology and medicine*, vol. 154, p. 106590, 2023.

[83] J. Li, D. Han, Z. Wu, J. Wang, K.-C. Li, and A. Castiglione, "A novel system for medical equipment supply chain traceability based on alliance chain and attribute and role access control," *Future Generation Computer Systems*, vol. 142, pp. 195–211, 2023.

[84] T. Hukkinen, J. Mattila, K. Smolander, T. Seppala, and T. Goodden, "Skimping on gas–reducing ethereum transaction costs in a blockchain electricity market application," 2019.

[85] K. Nelaturu, S. M. Beillahi, F. Long, and A. Veneris, "Smart contracts refinement for gas optimization," in *2021 3rd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS)*. IEEE, 2021, pp. 229–236.

[86] B. Gao, S. Shen, L. Shi, J. Li, J. Sun, and L. Bu, "Verification assisted gas reduction for smart contracts," in *2021 28th Asia-Pacific Software Engineering Conference (APSEC)*. IEEE, 2021, pp. 264–274.

[87] Y.-w. Jeng, Y.-c. Hsieh, and J.-L. Wu, "Step-by-step guidelines for making smart contract smarter," in *2019 IEEE 12th Conference on Service-Oriented Computing and Applications (SOCA)*, 2019, pp. 25–32.

[88] A. Di Sorbo, S. Laudanna, A. Vacca, C. A. Visaggio, and G. Canfora, "Profiling gas consumption in solidity smart contracts," *Journal of Systems and Software*, vol. 186, p. 111193, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0164121221002697

[89] K. Kaur, S. Tomar, and M. Tripathi, "Gas fee reduction by detecting loop fusible patterns in ethereum smart contract," in *2022 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, 2022, pp. 458–463.

[90] S. Farokhnia and A. Kafshdar Goharshady, "Alleviating high gas costs by secure and trustless off-chain execution of smart contracts," in *Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing*, ser. SAC '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 258–261. [Online]. Available: https://doi.org/10.1145/3555776.3577833

[91] R. Belchior, A. Vasconcelos, S. Guerreiro, and M. Correia, "A survey on blockchain interoperability: Past, present, and future trends," *ACM Computing Surveys (CSUR)*, vol. 54, no. 8, pp. 1–41, 2021.

[92] S. Schulte, M. Sigwart, P. Frauenthaler, and M. Borkowski, "Towards blockchain interoperability," in *Business Process Management: Blockchain and Central and Eastern Europe Forum: BPM 2019*

Blockchain and CEE Forum, Vienna, Austria, September 1–6, 2019, Proceedings 17. Springer, 2019, pp. 3–10.

[93] P. Lafourcade and M. Lombard-Platet, "About blockchain interoperability," *Information Processing Letters*, vol. 161, p. 105976, 2020.

[94] ——, "About blockchain interoperability," *Information Processing Letters*, vol. 161, p. 105976, 2020.

[95] T. Hardjono, A. Lipton, and A. Pentland, "Toward an interoperability architecture for blockchain autonomous systems," *IEEE Transactions on Engineering Management*, vol. 67, no. 4, pp. 1298–1309, 2019.

[96] G. Wang, Q. Wang, and S. Chen, "Exploring blockchains interoperability: A systematic survey," *ACM Computing Surveys*, 2023.

[97] K. Ren, N.-M. Ho, D. Loghin, T.-T. Nguyen, B. C. Ooi, Q.-T. Ta, and F. Zhu, "Interoperability in blockchain: A survey," *IEEE Transactions on Knowledge and Data Engineering*, 2023.

[98] J. Li, D. Han, Z. Wu, J. Wang, K.-C. Li, and A. Castiglione, "A novel system for medical equipment supply chain traceability based on alliance chain and attribute and role access control," *Future Generation Computer Systems*, vol. 142, pp. 195–211, 2023.

[99] M. Al Amin, A. Altarawneh, and I. Ray, "Informed consent as patient driven policy for clinical diagnosis and treatment: A smart contract based approach," in *Proceedings of the 20th International Conference on Security and Cryptography-SECRYPT*, 2023, pp. 159–170.

[100] Y. Nishat and S. Shahid, "Effects of smart-contract blockchain on smart healthcare center management," in *Unleashing the Potentials of Blockchain Technology for Healthcare Industries*. Elsevier, 2023, pp. 107–136.

[101] R. W. Ahmad, K. Salah, R. Jayaraman, I. Yaqoob, M. Omar, and S. Ellahham, "Blockchain-based forward supply chain and waste management for covid-19 medical equipment and supplies," *Ieee Access*, vol. 9, pp. 44 905–44 927, 2021.

[102] D. Hawashin, K. Salah, R. Jayaraman, I. Yaqoob, and A. Musamih, "A blockchain-based solution for mitigating overproduction and underconsumption of medical supplies," *IEEE Access*, vol. 10, pp. 71 669–71 682, 2022.

An Approach Based on Named Entity Recognition and Semantic Analysis for Recruitment Efficiency and Optimization

Ismail Ifakir[1], Noureddine Mohtaram[2], El Habib Nfaoui[3], Abderrahim Zannou[4], Mohammed El Hassouni[5]
L3IA Laboratory, Faculty of Sciences Dhar El Mahraz, Sidi Mohamed Ben Abdellah University, Fez, Morocco[1,3]
LRIT Rabat IT center, Faculty of Sciences, Mohammed V University in Rabat, Morocco[2]
ERC2IA, Faculty of Sciences and Techniques, Al Hoceima, Abdelmalek Essaadi University, Tetouan, Morocco[4]
FLSH, Mohammed V University in Rabat, Morocco[5]

***Abstract*—Modern recruitment requires smarter, faster, and more inclusive methods to manage the growing volume and diversity of job applications and candidate resumes. Manual screening is often ineffective and unreliable especially in low-resource or multilingual contexts. To address this challenge, we propose an approach that automates and optimizes key stages of the recruitment process. This three-stage approach includes: 1) extracting structured data from resumes using a robust Named Entity Recognition (NER) system, which comprises a NER annotator, a feature extractor, and a transition-based parser; 2) employing a fine-tuned transformer model to perform semantic matching between candidates and job descriptions; and 3) leveraging a large language model to generate interview questions tailored to specific job requirements, thereby improving the relevance and personalization of candidate assessments. The recruitment system was tested on a large-scale resume and job posting dataset across multiple domains. Our NER model reported an F1-score of 85.11% in entity extraction, and the matching component reported accuracy levels as high as 92% when using hierarchical job classes. The results prove the efficacy of combining deep learning techniques with semantic reasoning in enhancing automation, accuracy, and fairness in hiring.**

***Keywords*—Named entity recognition; large language models; feature extraction; generate question; matching**

I. Introduction

With technology advancing rapidly and professional fields becoming increasingly diverse, companies and recruitment systems are now confronted with an unprecedented volume of job applications. Large firms and headhunters process hundreds of Curriculum Vitae (CVs) each day, with each document exhibiting unique formatting styles, layouts, and structures. This variability poses a significant challenge for automated recruitment systems, which must efficiently extract, organize, and evaluate candidate information at scale. Named Entity Recognition (NER), a subtask of Natural Language Processing (NLP), offers one of the most promising solutions to this challenge. NER [1], [2], [3], [4], [5], [6], [7], [8], [9] refers to the process of identifying and classifying named entities in text into predefined categories including names, organizations, locations, dates, and other specific terms. NER has shown notable success in extracting structured knowledge from unstructured resume content [10], [11], [12], [13], [14], [15], in the framework of resume analysis. It helps important components name, contact information, educational background, professional experience, technical and soft skills, and certifications to automatically detect and organize. Another crucial component of intelligent recruitment is the task of matching candidates with job descriptions [16], [17], [18], [19], [20], [21], which aims to automatically align candidate profiles with employer requirements. The objective is to assess how well a candidate's qualifications, background, and skills correspond to the expectations and preferences outlined in job postings. Automating this matching process can significantly enhance the quality of candidate selection, reduce time-to-hire, and improve overall recruitment efficiency. However, this remains a complex and inherently challenging task. It goes beyond simply identifying similar keywords or job titles; it requires an understanding of the semantic context in both resumes and job descriptions. Terminologies often differ, experiences may only partially align, and important qualifications may be implied rather than explicitly stated. As a result, traditional keyword-based methods are frequently inadequate. To achieve accurate and meaningful matching, advanced techniques such as language embeddings, semantic similarity modeling, and context-aware entity alignment are increasingly being employed. Following the candidate–job matching phase, companies often need to conduct personalized evaluations of shortlisted applicants. At this stage, recruiters must delve deeper into specific aspects of each profile, which typically involves asking targeted follow-up questions. However, many organizations lack the time and resources to manually craft relevant questions for every candidate. To address this, the automatic generation of entity-driven interview questions has emerged as a valuable solution [22]. By leveraging entities extracted from resumes and job descriptions during the matching phase, the system can generate tailored questions for each candidate, helping recruiters verify key information or probe specific skills and experiences. For example, based on the entity work experience, the system might generate questions such as: Can you elaborate on your responsibilities at Company X? or What challenges did you face during your role as a Data Analyst? Thus enabling a more focused, consistent, and efficient evaluation process.

To address these challenges, we propose an approach that automates and optimizes key stages of the recruitment process. The process begins with the extraction of structured information from resumes using a robust Named Entity Recognition framework, composed of an annotator, a feature extraction, and a transition-based parser. This is followed by the use of a fine-tuned transformer-based model to achieve semantic

alignment between candidate profiles and job descriptions. Finally, leveraging a large language model to generate interview questions tailored to specific job requirements. This recruitment process mitigates the limitations of traditional recruitment workflows by automating resumes parsing, CVs-job matching, and question generation ultimately reducing manual effort, improving selection accuracy, and enhancing the overall efficiency of the recruitment process for both employers and applicants.

This study is structured as follows generally: Section II discusses related work, stressing developments in entity extraction from CVs and other matching techniques applied in recruitment. Section III covers the preliminaries and details on the models applied in this work. The suggested framework is presented in Section IV together with information on extracting important entities, matching between candidates and job descriptions, and entity-based interview questions. Section V addresses the experimental results, evaluating the performance of the framework in relation to current models and so analyzing its possible influence and performance. Section VI ends the work by aggregating important contributions and suggests future avenues of research to improve recruiting automation.

II. Related Work

In the field of CV parsing, many companies process large volumes of resumes to identify suitable candidates for employment, making manual analysis increasingly impractical. As a result, significant progress has been made in Named Entity Recognition for extracting structured information from resumes [23], [24], as well as in developing algorithms that efficiently match candidates with job requirements. Several approaches have been proposed to tackle these problems. For instance, Pham et al. [25] proposed a deep learning-based NER model for resume parsing in their paper "Study of Information Extraction in Resume". Their method involves four main steps: text normalization, rule-based NER, which based on deep neural networks, and text segmentation. Using a 1,000 manually labeled Vietnamese resume medium-sized dataset, they extracted entities such as names, addresses, talents, degrees, experience, and universities. Zhu et al. [26] proposed a user behavior-based preference alignment framework for fine-tuning LLMs to improve job recommendations through resume completion. To address noise in behavioral data (i.e., bias and variance), they introduced a noise-robust LLM alignment method named Denoised Direct Preference Optimization (Denoised DPO), which disentangles genuine user preferences from noisy data. Specifically, they designed a novel reward function for preference estimation by combining an LLM-based component for real user preference with a regression model for bias disentanglement. In addition, they developed a Thurstonian-style model for job-seekers' preference modeling to stabilize data reliability amidst behavior variances. Extensive offline and online experiments demonstrated the effectiveness of this approach in enhancing recommendation quality.

Vanetik and Kogan [27] improved resume-job matching using semantic similarity by taking the assistance of BERT-based sentence embeddings. They extracted entities like job titles, credentials, and skills using SpaCy's NER component, and employed keyword extraction using TF-IDF. The integration of keyword-based methods, BERT embeddings, and NER formed a solid base for automatic candidate ranking. Bakliwal et al.[28] addressed the problem of orphan entity resolution using a knowledge graph-based approach and models such as BERT and RoBERTa. Their model, via concept mining, association mining, and NER, detected and contextualized doubtful resume entries. The inferred knowledge graphs linked these entities to external knowledge bases, improving contextual comprehension and candidate-job matching. Wang et al. [29] examined the impact of pretrained language models, including BERT, ERNIE, ERNIE2.0-tiny, and RoBERTa, on NER performance. By fine-tuning all the models, they evaluated the effect of different pretraining strategies on extraction of entities from unstructured text and offered valuable information on model selection for NER applications. Tran et al. [30] proposed a practical solution to occupational skill identification in Vietnamese job advertisements in a bid to address the skill mismatch most prevalent in the domestic labor market. Rather than approach skill detection as a standard NER task, they approached it as ranking. Identified phrases were ranked by semantic similarity with context neighbors without involving extensive annotated datasets or supervised NER models. Rosenberger et al. [31] introduced CareerBERT, a deep learning-based model for resume-to-job matching enhancement. The model acquires the ability to update and adapt to dynamic labor market shifts using unstructured resume and job posting data. CareerBERT integrates ESCO taxonomy and EURES job posting data into an active and structured job corpus. The accuracy and validity of recommended jobs are improved as a consequence. Tyagi et al. [32] countered gendered bias in resume matching systems using word embedding refinement and improving classification algorithms. Their system demonstrates how gendered words, descriptions, and skill mentions can produce biased representations in resumes that result in unequal job classification. Using real-world datasets, they demonstrated high correlation between embedding bias and occupational gender skewing and proposed debiasing techniques to decrease the bias. Lastly, Barducci et al. [33] addressed the problem of resume information extraction in the Italian labor market by proposing an effective end-to-end framework. Their system provides a complete candidate overview, including personal information, skills, and work experiences. The framework extracts raw data from resumes and segments them into semantically consistent parts using linguistic patterns. Each segment is then processed with a NER algorithm, based on pre-trained language models, to extract the most relevant information.

Previous studies have demonstrated a wide range of approaches aimed at addressing entity extraction challenges in resume parsing and job matching. While substantial progress has been made such as generating evaluation questions from resumes, identifying resumes that closely match specific job descriptions, and aligning job postings with suitable candidates significant gaps remain. These include the underutilization of advanced deep learning models, limited integration of semantic embeddings, and insufficient application of knowledge graphs for contextual understanding. To address these limitations, this study proposes a more comprehensive approach to entity extraction from resumes using a NER-based framework. The goal is to enable more accurate and context-aware parsing of

candidate profiles, facilitate scalable and effective matching between resumes and job descriptions, and automatically generate relevant interview questions based on extracted entities.

III. Preliminaries

This section provides the necessary background to support the rest of the study, including named entity recognition, text embeddings, pre-trained transformer-based models from the BERT family, and large language models.

A. Named Entity Recognition

Named Entity Recognition [1] is a core task in natural language processing that involves identifying and classifying entities in text into predefined categories such as persons, organizations, locations, dates, and others. NER systems typically rely on sequence labeling techniques, where each token is assigned a tag indicating whether it belongs to an entity and what type it represents. NER is widely applied in various real-world scenarios, including information extraction from unstructured text, question answering, content classification, knowledge graph construction, and automated resume analysis. In the recruitment domain, for example, NER is essential for extracting structured information from resumes and job descriptions, such as candidate names, educational qualifications, professional experience, and specific skills. Modern NER approaches leverage contextual embeddings and transformer-based architectures to achieve high accuracy and robustness, particularly in domain-specific or noisy text environments.

B. Embeddings

Embeddings are numerical models of unprocessed data text or category values, for example into continuous vector spaces. Basic to deep learning models, reflecting semantic linkages and similarities between data points unlike traditional discrete or one-hot representations is not a secret. Mostly for the preservation of pertinent information, embeddings convert high-dimensional data into dense, lower-dimensional representations. This ensures more successful and efficient application for jobs involving classification, grouping, and similarity analysis. By grouping semantically or structurally similar objects closer together in the embedding space, embeddings let machine learning algorithms find underlying patterns in the data and overcome expressly specified constraints.

C. BERT Family

Processing text bidirectionally, the deep learning model BERT (Bidirectional Encoder Representations from Transformers) [34] considers both left and right contexts concurrently. It discovers contextual links between words inside a sentence by means of Transformer architecture and more importantly, self-attention methods. For many NLP uses, large corpus pre-training helps BERT provide highly contextualized embeddings. In fields including Named Entity Recognition, it has greatly increased performance, sometimes exceeding historical benchmarks. By means of fine-tuning, BERT can be customized to domain-specific tasks even with limited labeled data.

Under the Masked Language Modeling aim, Bert is trained whereby the model is taught to forecast random words in a phrase by masking them. Moreover, next sentence prediction (NSP) improves sentence-level comprehension. Though more contemporary models like RoBERTa eliminate the NSP aim, BERT's multi-layer Transformer encoders capture deep language patterns, making it quite well-suited for NER, information extraction, and automated text processing applications. Dynamic masking eliminates the NSP aim and produces a superior variant of BERT, RoBERTa, thereby, enhancing the pretraining process. Unlike BERT's fixed masking, RoBERTa exposes the model to a wider diversity of linguistic settings, as masking patterns vary on the same input across training epochs. From this, in NER, resulting more solid representations and better task performance. RoBERTa has shown more accuracy and efficiency in named entity extraction, however its architecture is essentially similar BERT.

By means of disentangled attention via DeBERTa (Decoding-enhanced BERT with disentangled attention), one obtains important benefits by separating content and spatial embeddings. While traditional models mix lexical meaning and spatial information, DeBERTa especially separates these elements thereby improving the model's grasp of word order and sentence structure. This method shows especially good performance in entity recognition tasks, where exact capture of subtle dependencies is essential. DeBERTa also features a new decoding technique designed to raise token prediction accuracy.

More recently, CamemBERT a multilingual BERT-based model tailored for the labor market domain, has shown strong performance in skill and occupation extraction tasks. It is trained using job-related taxonomies and multilingual job corpora to enhance cross-lingual generalization and semantic understanding in employment-related text. By specializing in the semantics of job postings and CVs, CamemBERT proves particularly effective for both skill detection and entity recognition in multilingual or non-standard CV formats.

D. Large Language Models

LLMs are advanced neural network architectures trained on massive text corpora to understand and generate human-like language. Built primarily on transformer architectures, LLMs utilize self-attention mechanisms to model complex linguistic patterns, enabling them to perform a wide range of natural language processing tasks such as summarization, translation, question answering, and dialogue generation. Their ability to generate contextually relevant and coherent responses makes them suitable for integration into applications requiring intelligent language understanding and generation. Notable examples of LLMs include Google Gemini developed by Google DeepMind for advanced language understanding and generation, and Meta's LLaMA (Large Language Model Meta AI) designed as an efficient, open-weight model for research and practical applications. These models represent key milestones in the evolution of generative AI and have significantly contributed to progress across a variety of domains.

IV. Proposed Approach

This section describes the method adopted to enhance and automate the recruitment process. The approach leverages NER for entity extraction, candidate–job matching, and entity-based question generation.

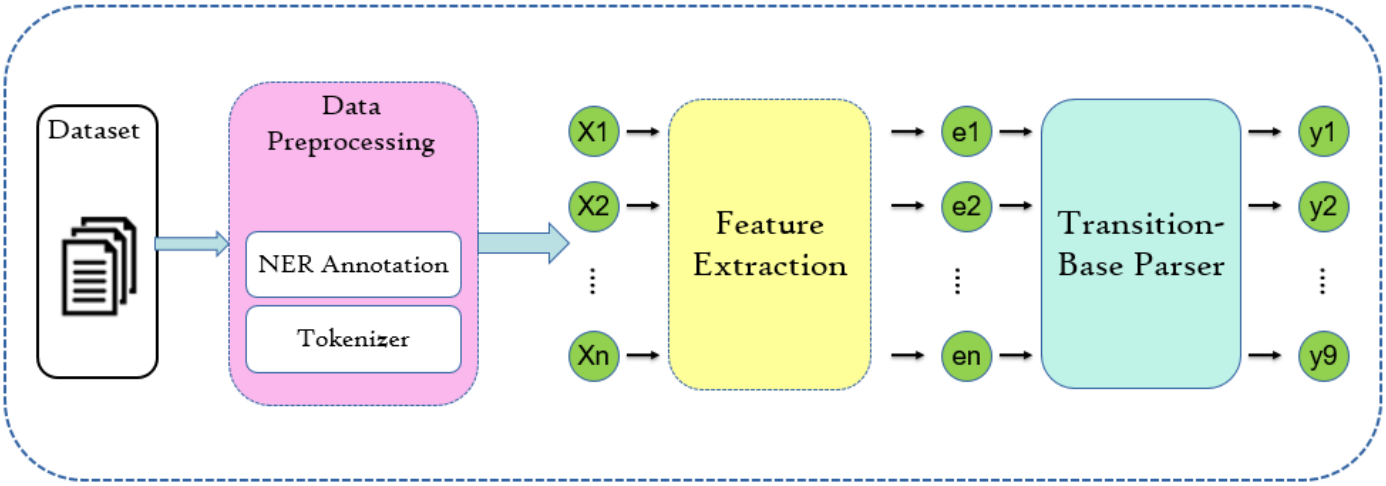


Fig. 1. Overview of the proposed NER pipeline architecture.

A. NER-Based Entity Extraction

The structure of the NER entity extraction model is illustrated in Fig. 1. This framework is composed of five key modules: data collection, preprocessing, feature extraction, and the transition-based parser. To effectively leverage these components for feature learning and entity recognition, the process begins with model representation. This involves collecting a dataset of resumes in various formats and converting them into machine-readable text. The preprocessing stage consists of two essential tasks: Named Entity Recognition annotation and tokenization. Following that, a pre-trained transformer model processes the tokens to generate contextual embeddings that capture semantic relationships between words. Finally, the transition-based parser applies a sequence of learned transitions to progressively predict entity boundaries and assign labels, enabling the extraction of structured entities from unstructured resume text.

1) Data preprocessing: From multiple sources including a broad spectrum of companies, job vacancies, and skill levels we assembled a diversified collection of resumes in PDF and Word formats. These resumes are written in English and follow a consistent structure, enabling uniform data representation. We denote the dataset as $D = \{d_1, d_2, \ldots, d_n\}$, where D represents the collection of n resumes, and each d_i is a raw document. For more details about the dataset, refer to Section V.

Each document d_i in the dataset is first manually or semi-automatically annotated with named entities directly on the raw text. These annotations are created as labeled character spans and stored as triplets: $A_i = \{(s_k, e_k, l_k)\}$. Here, s_k and e_k denote the character-level start and end offsets of an entity, and l_k is the corresponding entity label (e.g., `AGE`, `SKILLS`).

After annotation, the raw text is passed through spaCy's rule-based tokenizer, which segments it into a sequence of tokens: $T_i = \{t_{i1}, t_{i2}, \ldots, t_{im}\}$. During training, the character-based entity spans A_i are automatically aligned with the token boundaries defined in T_i, producing token-level supervision for the NER model.

2) Contextual embedding: Once tokenized, the sequence T_i is passed into a pre-trained transformer model such as RoBERTa, via the spaCy-transformers integration. This model generates a contextual embedding for each token:

$$E_i = \text{Em}(T_i) = \{\mathbf{e}_{i1}, \mathbf{e}_{i2}, \ldots, \mathbf{e}_{im}\}, \quad \mathbf{e}_{ij} \in \mathbb{R}^d \tag{1}$$

These contextual embeddings in Eq. (1) encode each token with rich semantic and syntactic information by leveraging sentence-level dependencies and multi-head attention mechanisms.

3) Transition-based parser: The final step is handled by the Transition-Based Parser module, which consumes the token vectors and predicts entity boundaries and labels through a sequence of learned transitions. At each state s, the model scores and selects the most probable action a_s, as shown in Eq. (2):

$$a_s = \arg\max_{a \in A} f_{\text{score}}(\sigma_s, a) \tag{2}$$

where, σ_s is the parser state, A is the set of possible transition actions (e.g., BEGIN, CONTINUE, OUT), and f_{score} is the internal scoring function (a feed-forward network with maxout layers). The final labeled spans are stored in `doc.ents` for downstream use.

B. Matching

Fig. 2 illustrates the overall architecture of the matching process between candidates and job descriptions. It consists of four main components: Input Data, Entity Extraction, Feature Extraction, and Similarity Matching. The process begins with unstructured input data, namely job descriptions and candidate resumes written in French.

In the first step, entities are extracted using a custom NER model, as detailed in Fig. 1. This model identifies key elements such as skills, technologies, and job-specific attributes

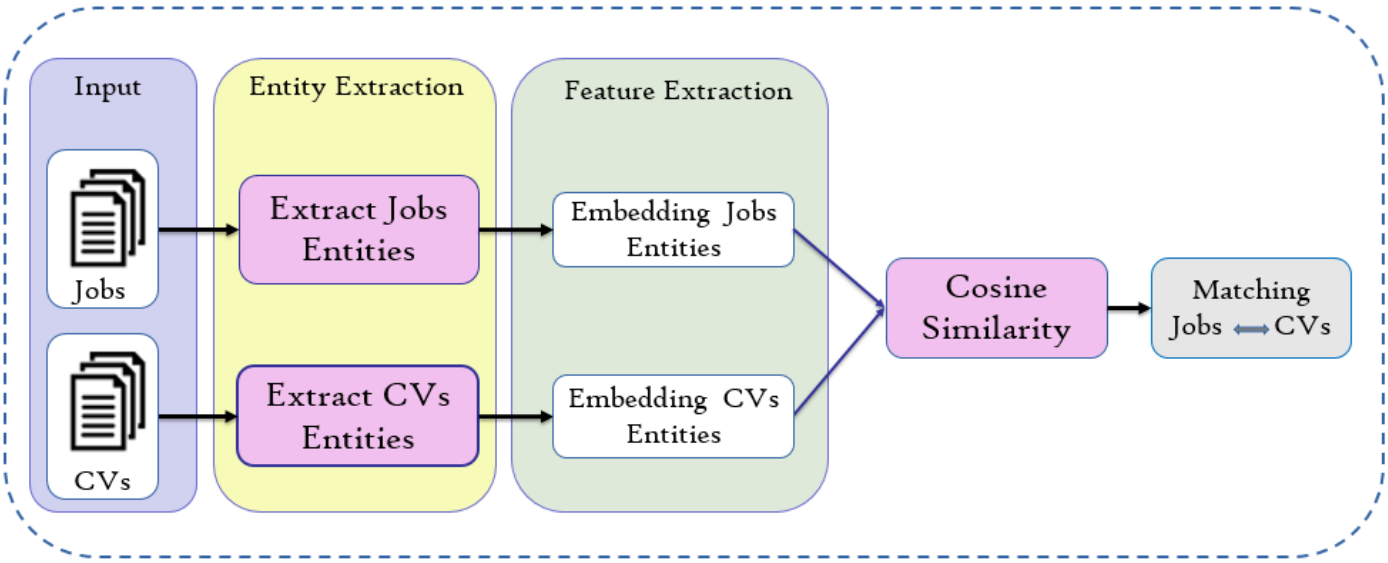


Fig. 2. Overview of the matching framework.

from both the job descriptions and the CVs. These extracted entities are then passed to a feature extraction module, which employs a pre-trained transformer model to produce contextual embeddings. Finally, cosine similarity is computed to assess the semantic alignment between each job description and the resumes. If the similarity score exceeds a given threshold, the corresponding CV is considered a potential match.

1) Input: The system takes as input a job description $J = \{J_1, J_2, \ldots, J_n\}$ and a set of resumes $R = \{R_1, R_2, \ldots, R_n\}$, where each J_i and R_i consists of unstructured text written in French. The input for each comparison is formally defined as Input $= (J_i, R_i)$, representing the pairing between a job description and a candidate resume.

2) Entity extraction: Entities such as skills, technologies, and job-related information are extracted from both the job description and the resumes using a Named Entity Recognition (NER) model F. For each pair, the model is applied as $S_{J_i} = F(J_i)$ and $S_{R_i} = F(R_i)$, where S_{J_i} denotes the set of entities extracted from the job description J_i, and S_{R_i} corresponds to the entities extracted from the i-th resume R_i, for all $i \in \{1, \ldots, n\}$.

3) Feature extraction: The extracted entities are then passed through a CamemBERT-based transformer model which serves as the embedding function Em to generate dense vector representations [see Eq. (3)]:

$$U = Em(S_{J_i}), \quad V_i = Em(S_{R_i}) \quad \forall i \tag{3}$$

where, $U \in \mathbb{R}^d$ is the embedding vector for the job description, and $V_i \in \mathbb{R}^d$ represents the embedding vector for the i-th resume.

4) Similarity: To measure the alignment between a job description and a resume, cosine similarity is computed between their corresponding embeddings, as defined in Eq. (4):

$$\text{sim}(U, V_i) = \frac{U \cdot V_i}{\|U\| \cdot \|V_i\|}, \quad \forall i \tag{4}$$

A resume is considered a match if the similarity in Eq. (5) exceeds a predefined threshold:

$$\text{sim}(U, V_i) \geq \tau \tag{5}$$

where, $\tau = 0.5$ denotes the similarity threshold.

This similarity-based matching approach between candidates and job descriptions, as formulated in Eq. (5), enables a more accurate and efficient recruitment process.

C. Entity-Based Question Generation

After a match is confirmed between candidates and job descriptions, the process triggers a Large Language Model to generate relevant interview questions. For this task, we evaluated two LLMs: Google Gemini and Meta's LLaMA. While both models are capable of producing grammatically correct and contextually appropriate questions, Google Gemini demonstrated superior performance in terms of relevance and alignment with the extracted entities.

Gemini's advantages lie in its advanced instruction tuning, stronger contextual reasoning, and ability to generate coherent and targeted questions without the need for additional fine-tuning. On the other hand, LLaMA-based models often require instruction-tuned variants to achieve comparable results.

Therefore, Google Gemini is preferred in our pipeline due to its robustness, high-quality output, and ease of integration.

V. Experimental Results and Evaluation

We assessed in this work several approaches and models for entity extraction and matching between candidates and job descriptions.

A. Extract Entities

In this subsection, we first introduce the dataset used in the experiments. We then describe the experimental setup for both NER-based entity extraction model and the baseline models. Finally, we present a series of experimental results to validate the effectiveness of the proposed model.

1) Data description: We performed entity extraction to improve hiring efficiency in recruitment systems by applying Named Entity Recognition to the task of parsing and analyzing resumes. To support this, we compiled a dataset comprising both structured and semi-structured resume documents. The dataset, stored in .json format, included a total of 1,000 resumes. Of these, 800 were used for training and 200 for testing. This 80/20 split was adopted to ensure a reliable assessment of the model's ability to generalize to unseen data.

Each resume was annotated with key entities relevant to recruitment and candidate evaluation. These included: Name, LinkedIn Link, Location, Degree, College Name, Email Address, University Name, Skills and Certifications. These entities were carefully chosen to reflect the most critical information recruiters seek during the hiring process. The dataset covered a broad range of resume formats and writing styles, thereby improving the robustness and generalization capabilities of the NER models.

2) Implementation setup: We evaluate the proposed entity extraction method to improve hiring efficiency in recruitment systems by applying Named Entity Recognition to the dataset described earlier. This dataset provides a solid foundation for assessing the performance of various text-based models under different hyperparameter configurations.

Our model was trained using the hyperparameters in Table I. Specifically, we used RoBERTa as our underlying transformer for feature extraction due to its superior performance on language understanding tasks. 0.1 was the dropout rate that we utilized in order to reduce the probability of overfitting by randomly eliminating neurons when training. We employed the Adam optimizer, where learning rates are adapted dynamically for all parameters, and set the initial learning rate to 5×10^{-5} in order to achieve steady and smooth convergence.

TABLE I. SUMMARY OF MODEL HYPERPARAMETERS AND DETAILS

Category	Value
Evaluation Metrics	F1-Score, Precision, Recall
Feature Extraction	RoBERTa
Dropout Rate	0.1
Optimizer	Adam
Learning Rate	5×10^{-5}
Early Stopping	1600 steps
GPU Allocator	PyTorch
Max Steps	20

For monitoring training and preventing overfitting, we employed early stopping with patience 1600 steps, i.e., training would be halted if improvement in validation loss was not observed within that time. Training was conducted for a maximum of 20 steps (or epochs), with adequate iterations for the model to learn without incurring high computational expenses. Training was conducted in PyTorch with GPU acceleration, providing efficient use of resources through dynamic memory allocation and fast batch processing.

For evaluating the performance of the model, we employed typical NER evaluation metrics: Precision, which provides an estimate of the accuracy of the entity predictions; Recall, which provides an estimate of the capability of the model to capture all entities of interest; and F1-score, the harmonic mean of recall and precision. These estimates provide an overall estimate of the capability of the model to correctly and comprehensively extract entity information from resumes.

3) Model variations: To assess the performance of various transformer-based architectures for Named Entity Recognition (NER) in the context of resume entity extraction, experiments were conducted using three widely adopted pre-trained models: BERT, DeBERTa, and RoBERTa, each employed as a feature extractor. All models were fine-tuned under identical training conditions to ensure a fair comparison. Their performance was evaluated across multiple entity types using standard evaluation metrics, including Precision, Recall, and F1-score.

Table II presents the relative performance of the three models across ten target entity types. All results are listed in percentages (%) for each measurement. RoBERTa generally outperformed both BERT and DeBERTa for nearly all categories, achieving the highest F1-scores in major domains such as Name, LinkedIn Link, Location, Degree, College Name, Email Address, University Name, Skills and Certifications, and also in the overall score for all entities

RoBERTa achieved a combined F1-score of 85.11%, outperforming DeBERTa (82.94%) and BERT (80.14%). This improvement is a witness to the enhanced contextual depth understanding capacity of RoBERTa for resume content. The model performed strongly in proper categories like Email Address and Skills, which are central to hiring decision-making.

While BERT was a decent baseline, it lagged behind in recall for many entity types, indicating a bias towards missing relevant entities. DeBERTa had better recall than BERT, especially in categories such as Email Address and Location, but was less dependable overall than RoBERTa.

These findings confirm that choice of transformer architecture exerts a significant impact on the effectiveness of NER-based resume parsing. Of the models tested, RoBERTa emerged as the most powerful and most reliable and therefore an infinitely feasible option to use in automated hiring processes.

B. Matching Evaluation

In this part of our analysis, we evaluate the performance of the candidates–jobs matching component. We begin by introducing the similarity-based scoring framework used to quantify the relevance between candidates and recruiters. We then present a summary of the matching performance, including evaluation metrics and comparative analysis.

1) Similarity-based scoring framework: To assess the effectiveness of our matching system, we use a similarity-based scoring approach. When a CV or a job description is entered,

TABLE II. COMPARISON OF OUR NER RESULTS ON RESUMES (%)

Entities	Feature Extraction								
	BERT			DeBERTa			RoBERTa		
	F1-Score	Precision	Recall	F1-Score	Precision	Recall	F1-Score	Precision	Recall
All Entities	80.14	80.43	81.91	82.94	83.61	83.37	**85.11**	**85.24**	**86.47**
Name	79.09	79.34	80.57	81.65	83.86	79.66	**85.20**	**85.06**	**87.56**
LinkedIn Link	78.95	79.31	79.33	77.52	83.95	71.73	**84.75**	**84.71**	**87.79**
Location	79.24	78.17	80.34	81.91	83.14	82.28	**84.69**	**84.86**	**86.47**
Degree	78.49	77.67	79.46	81.44	83.26	80.52	**85.28**	**84.47**	**87.02**
College Name	78.81	78.14	79.50	78.09	83.12	74.83	**84.82**	**84.36**	**85.88**
Email Address	80.37	79.10	81.70	84.85	84.11	85.59	**85.54**	**85.63**	**87.33**
University Name	78.92	77.43	80.46	81.52	83.10	80.00	**84.90**	**85.19**	**85.78**
Skills	80.01	79.89	80.57	84.22	84.94	84.65	**85.03**	**84.94**	**86.83**
Certifications	78.04	76.57	79.58	78.90	83.52	75.60	**85.20**	**84.82**	**87.08**

the system calculates a semantic similarity score between them using embeddings generated by the CamemBERT model. The evaluation pipeline is detailed below:

a) Entity embedding for job description and CVs: Let T denote the job description text and C_i represent the entities (e.g., skills) extracted from the i-th CV. The embeddings for both are obtained via the CamemBERT model, as defined in Eq. (6):

$$E_T = \mathrm{E}_m(T), \quad E_{C_i} = \mathrm{E}_m(\text{Entities}_i) \tag{6}$$

where, T denotes the job description text, Entities_i represents the set of extracted skills from the i-th candidate's CV, E_T is the embedding vector corresponding to the job description, and E_{C_i} refers to the embedding of the entities extracted from the i-th CV.

b) Similarity computation using cosine metric: The similarity score KM_i between the job description and each CV is calculated using cosine similarity [Eq. (7)]:

$$\text{KM}_i = \frac{E_T \cdot E_{C_i}}{\|E_T\| \|E_{C_i}\|} \tag{7}$$

where, $E_T \cdot E_{C_i}$ denotes the dot product between the job description and the candidate CV embeddings, and $\|E_T\|$ and $\|E_{C_i}\|$ are the Euclidean norms of the respective embedding vectors.

c) Score normalization: To ensure consistent scoring across different inputs, the similarity scores are normalized using their mean μ and standard deviation σ, as given in Eq. (8):

$$\mu = \frac{1}{n}\sum_{i=1}^{n} \text{KM}_i, \quad \sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\text{KM}_i - \mu)^2} \tag{8}$$

If $\sigma = 0$, it is replaced with $\sigma = 1$ to avoid division by zero. The normalized difference is then computed [Eq. (9)]:

$$\text{Diff}_i = \frac{\text{KM}_i - \mu}{\sigma} \tag{9}$$

d) Rating conversion: The normalized score is mapped on a 0 to 10 rating scale using the transformation in Eq. (10):

$$\text{Rating}_i = \min\left(10, \max\left(0, 5 + \text{Diff}_i\right)\right) \tag{10}$$

This ensures that:

- Ratings remain within the range [0, 10].
- Ratings are centered around a neutral baseline of 5.

e) Matching accuracy metric: Finally, the matching accuracy is computed as the average rating of the top k most similar CVs, as shown in Eq. (11):

$$\text{Accuracy} = \frac{1}{k}\sum_{i=1}^{k} \text{Rating}_i \tag{11}$$

In our evaluation, $k = 5$ was used.

2) Summary of matching results: The matching accuracy was evaluated using the method described in the previous section (Accuracy Computation Method). The result reflects the average performance of the top five matched CVs, based on cosine similarity between job descriptions and resume embeddings.

The evaluation shows that the CamemBERT-based model achieves strong performance, with an accuracy of 92%. This highlights the effectiveness of the proposed semantic matching pipeline in retrieving relevant candidate profiles from unstructured textual data.

These findings suggest that the proposed AI-based job matching system offers accurate candidate recommendations, efficient processing, and strong potential for deployment in real-world recruitment workflows.

VI. Conclusion and Future Work

In this work, we propose an approach that automates and optimizes the recruitment process. By combining entity extraction, candidates–jobs matching, and entity-based question generation, the approach offers a complete solution for streamlining the recruitment process. Leveraging natural language processing techniques and semantic similarity models, the process reduces manual effort, enhances candidate evaluation, and increases objectivity in decision making. One major challenge encountered is the handling of resumes that contain text embedded in images, often the case with visually designed resumes. This can lead to data loss and the unintended exclusion of qualified candidates. Addressing this limitation is crucial to ensure data integrity and maintain the system's effectiveness across varied recruitment contexts.

Future work could focus on incorporating OCR (Optical Character Recognition) technologies or image-to-text conversion models to enable the processing of a broader range of resume formats. Enhancing the system with real-time feedback mechanisms, improving its scalability for large-scale hiring scenarios, and expanding its adaptability across different industries would further increase its utility. Moreover, integrating context-aware decision-making and optimizing computational performance could elevate the overall intelligence and responsiveness of the system. These developments would contribute to building a more resilient and inclusive recruitment solution, delivering tangible benefits for both organizations and job seekers.

References

[1] A. Sharma, Amrita, S. Chakraborty, and S. Kumar, "Named entity recognition in natural language processing: A systematic review," in *Proceedings of Second Doctoral Symposium on Computational Intelligence: DoSCI 2021*, Springer, 2022, pp. 817–828. https://doi.org/10.1007/978-981-16-3346-1_66https://doi.org/10.1007/978-981-16-3346-1_66.

[2] Mansouri, A., Affendey, L. S., & Mamat, A. (2008). "Named entity recognition approaches." *International Journal of Computer Science and Network Security*, 8(2), 339–344. Citeseer. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=f3a467f1a32b844b3f04321212d49ee4f215d484doi:f3a467f1a32b844b3f04321212d49ee4f215d484.

[3] Li, J., Sun, A., Han, J., & Li, C. (2020). "A survey on deep learning for named entity recognition." *IEEE Transactions on Knowledge and Data Engineering*, 34(1), 50–70. IEEE. https://doi.org/10.1109/TKDE.2020.2981314https://doi.org/10.1109/TKDE.2020.2981314.

[4] Vychegzhanin, S., & Kotelnikov, E., "Comparison of Named Entity Recognition Tools Applied to News Articles," *Proceedings of the 2019 Ivannikov ISPRAS Open Conference (ISPRAS)*, pp. 72–77, 2019, IEEE. https://doi.org/10.1109/ISPRAS47671.2019.00017https://doi.org/10.1109/ISPRAS47671.2019.00017.

[5] P. Sun, X. Yang, X. Zhao, and Z. Wang, "An overview of named entity recognition," in *2018 International Conference on Asian Language Processing (IALP)*, IEEE, 2018, pp. 273–278. https://doi.org/10.1109/IALP.2018.8629225https://doi.org/10.1109/IALP.2018.8629225.

[6] X. Schmitt, S. Kubler, J. Robert, M. Papadakis, and Y. LeTraon, "A replicable comparison study of NER software: StanfordNLP, NLTK, OpenNLP, SpaCy, Gate," in *2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, IEEE, 2019, pp. 338–343. https://doi.org/10.1109/SNAMS.2019.8931850https://doi.org/10.1109/SNAMS.2019.8931850.

[7] H. Shelar, G. Kaur, N. Heda, and P. Agrawal, "Named entity recognition approaches and their comparison for custom ner model," *Science & Technology Libraries*, vol. 39, no. 3, pp. 324–337, 2020, Taylor & Francis. https://doi.org/10.1080/0194262X.2020.1759479https://doi.org/10.1080/0194262X.2020.1759479.

[8] C. Gayathri and R. S. Ravindran, "Named entity recognition using Bi-LSTM model with pointer cascade conditional random field for selecting high-profit products," *Egyptian Informatics Journal*, vol. 31, p. 100703, 2025. https://doi.org/10.1016/j.eij.2025.100703

[9] Y. Hu, Y. Chen, and Y. Xu, "A shape composition method for named entity recognition," *Neural Networks*, vol. 187, p. 107389, 2025. https://doi.org/10.1016/j.neunet.2025.107389

[10] A. Thukral, S. Dhiman, R. Meher, and P. Bedi, "Knowledge graph enrichment from clinical narratives using NLP, NER, and biomedical ontologies for healthcare applications," *International Journal of Information Technology*, vol. 15, no. 1, pp. 53–65, 2023, Springer. https://doi.org/10.1007/s41870-022-01145-yhttps://doi.org/10.1007/s41870-022-01145-y.

[11] Chantrapornchai, C., & Tunsakul, A., "Information Extraction Based on Named Entity for Tourism Corpus," *Proceedings of the 2019 16th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, pp. 187–192, 2019, IEEE. https://doi.org/10.1109/JCSSE.2019.8864166https://doi.org/10.1109/JCSSE.2019.8864166.

[12] S. Pudasaini, S. Shakya, S. Lamichhane, S. Adhikari, A. Tamang, and S. Adhikari, "Application of NLP for information extraction from unstructured documents," in *Expert Clouds and Applications: Proceedings of ICOECA 2021*, Springer, 2022, pp. 695–704. https://doi.org/10.1007/978-981-16-2126-0_54https://doi.org/10.1007/978-981-16-2126-0_54.

[13] Q. Zeng, M. Yuan, Y. Su, J. Mi, Q. Che, and J. Wan, "Improving Multimodal Named Entity Recognition via Text-image Relevance Prediction with Large Language Models," *Neurocomputing*, p. 130982, 2025. https://doi.org/10.1016/j.neucom.2025.130982

[14] Z. Zhou, L. Wei, and H. Luan, "Deep learning for named entity recognition in extracting critical information from struck-by accidents in construction," *Automation in Construction*, vol. 173, p. 106106, 2025. https://doi.org/10.1016/j.autcon.2025.106106

[15] X. Guo, Y. Chen, R. Tang, and Q. Zheng, "Camouflaged named entity recognition in 2D sentence representation," *Expert Systems with Applications*, vol. 257, p. 125096, 2024. https://doi.org/10.1016/j.eswa.2024.125096

[16] Guo, Shiqiang, Alamudun, Folami, & Hammond, Tracy, "RéSuMatcher: A Personalized Résumé-Job Matching System," *Expert Systems with Applications*, vol. 60, pp. 169–182, 2016, Elsevier. https://doi.org/10.1016/j.eswa.2016.04.013https://doi.org/10.1016/j.eswa.2016.04.013.

[17] Zaroor, A., Maree, M., & Sabha, M., "JRC: A Job Post and Resume Classification System for Online Recruitment," *Proceedings of the 2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI)*, pp. 780–787, 2017, IEEE. https://doi.org/10.1109/ICTAI.2017.00123https://doi.org/10.1109/ICTAI.2017.00123.

[18] Zaroor, A., Maree, M., & Sabha, M., "A Hybrid Approach to Conceptual Classification and Ranking of Resumes and Their Corresponding Job Posts," *Proceedings of the 9th KES International Conference on Intelligent Decision Technologies (KES-IDT 2017)–Part I*, pp. 107–119, 2018, Springer. https://doi.org/10.1007/978-3-319-59421-7_10https://doi.org/10.1007/978-3-319-59421-7_10.

[19] Martinez-Gil, J., Paoletti, A. L., & Pichler, M., "A Novel Approach for Learning How to Automatically Match Job Offers and Candidate Profiles," *Information Systems Frontiers*, vol. 22, pp. 1265–1274, 2020, Springer. https://doi.org/10.1007/s10796-019-09929-7https://doi.org/10.1007/s10796-019-09929-7.

[20] Sarveshwaran, R., Karthikeyan, S., Cruz, Meenalosini V., Shreyanth, S., Niveditha, S., & Rajesh, P.K., "NLP-Based AI-Driven Resume Screening Solution for Efficient Candidate Selection," *International Congress on Information and Communication Technology*, pp. 359–370, 2024, Springer. https://doi.org/10.1007/978-981-97-3556-3_29https://doi.org/10.1007/978-981-97-3556-3_29.

[21] A. Khatua and W. Nejdl, "Matching recruiters and jobseekers on Twitter," in *Proceedings of the 2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 2020, pp. 266–269. https://doi.org/10.1109/ASONAM49781.2020.9381392

[22] Y. Rebboud, L. Tailhardat, P. Lisena, and R. Troncy, "Can LLMs Generate Competency Questions?," in *Proceedings of the European Semantic Web Conference (ESWC)*, Springer, 2024, pp. 71–80. [Online]. Available: https://doi.org/10.1007/978-3-031-78952-6_7

[23] A. Gendrin, L. Souliotis, J. Loudon-Griffiths, R. Aggarwal, D. Amoako, G. Desouza, S. Dimitrievska, P. Metcalfe, E. Louvet, and H. Sahni, "Identifying patient populations in texts describing drug approvals through deep learning–based information extraction: Development of a natural language processing algorithm," *JMIR Formative Research*, vol. 7, p. e44876, 2023. https://doi.org/10.2196/44876

[24] Al-Moslmi, T., Gallofré Ocaña, M., Opdahl, A. L., and Veres, C., "Named entity extraction for knowledge graphs: A literature overview," *IEEE Access*, vol. 8, pp. 32862–32881, 2020. https://doi.org/10.1109/ACCESS.2020.2973928https://doi.org/10.1109/ACCESS.2020.2973928.

[25] Pham, V. L., Vu, N. S., & Nguyen, V. V. (2018). "Study of information extraction in resume." In Conference Proceedings. https://eprints.uet.vnu.edu.vn/eprints/id/eprint/3349/https://eprints.uet.vnu.edu.vn/eprints/id/eprint/3349/

[26] C. Zhu, X. Hu, H. Wu, C. Qin, H. Zhu, and H. Xiong, "Enhancing job recommendations with LLM-based resume completion: A behavior-denoised alignment approach," *Information Processing & Management*, vol. 62, no. 6, p. 104261, 2025. https://doi.org/10.1016/j.ipm.2025.104261

[27] Vanetik, N.; Kogan, G. "Job vacancy ranking with sentence embeddings, keywords, and named entities." *Information* **2023**, *14*, 468. https://doi.org/10.3390/info14080468https://doi.org/10.3390/info14080468.

[28] Bakliwal, A.; Gandhi, S. M.; Haribhakta, Y. "Leveraging Knowledge Graphs for Orphan Entity Allocation in Resume Processing." In *Proceedings of the 2023 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)*; IEEE: 2023; pp. 123–128. https://doi.org/10.1109/IICAIET59451.2023.10291293https://doi.org/10.1109/IICAIET59451.2023.10291293.

[29] Wang, Yu, Yining Sun, Zuchang Ma, Lisheng Gao, Yang Xu, and Ting Sun. "Application of pre-training models in named entity recognition." In *Proceedings of the 2020 12th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, vol. 1, pp. 23-26. IEEE, 2020. https://doi.org/10.1109/IHMSC49165.2020.00013https://doi.org/10.1109/IHMSC49165.2020.00013.

[30] Tran, Viet-Trung, Hai-Nam Cao, and Tuan-Dung Cao. "A practical method for occupational skills detection in Vietnamese job listings." In *Proceedings of the Asian Conference on Intelligent Information and Database Systems*, pp. 571-581. Springer, 2022. https://doi.org/10.1007/978-3-031-21743-2_46https://doi.org/10.1007/978-3-031-21743-2_46.

[31] Rosenberger, Julian, Lukas Wolfrum, Sven Weinzierl, Mathias Kraus, and Patrick Zschech. "CareerBERT: Matching resumes to ESCO jobs in a shared embedding space for generic job recommendations." *Expert Systems with Applications*, Elsevier, 2025, p. 127043. https://doi.org/10.1016/j.eswa.2025.127043https://doi.org/10.1016/j.eswa.2025.127043.

[32] Tyagi, Swati, Wei Qian, Jiaheng Xie, Rick Andrews, and others. "Enhancing gender equity in resume job matching via debiasing-assisted deep generative model and gender-weighted sampling." *International Journal of Information Management Data Insights*, vol. 4, no. 2, Elsevier, 2024, pp. 100283. https://doi.org/10.1016/j.jjimei.2024.100283https://doi.org/10.1016/j.jjimei.2024.100283.

[33] A. Barducci, S. Iannaccone, V. La Gatta, V. Moscato, G. Sperlì, and S. Zavota, "An end-to-end framework for information extraction from Italian resumes," *Expert Systems with Applications*, vol. 210, p. 118487, 2022. https://doi.org/10.1016/j.eswa.2022.118487

[34] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, vol. 1, pp. 4171–4186, Minneapolis, Minnesota, Jun. 2019. https://doi.org/10.18653/v1/N19-1423

Towards a Robust DNA Storage System with a Multilayer Approach

Ayoub Sghir, Manar Sais, Douha Bourached, Jaafar Abouchabaka, Najat Rafalia
Department of Computer Science, Faculty of Sciences, Ibn Tofail University, Kenitra, Morocco

***Abstract*—The growing demand for data storage requires innovative, resilient solutions to the challenges of cost, space, and energy consumption posed by current methods. DNA stands out as a promising next-generation data storage medium, offering a remarkable storage density of 1019 bits per cubic centimeter, some eight orders of magnitude denser than conventional media. This study explores the potential of DNA storage by proposing an intelligent multi-layer solution to overcome current technological challenges. The system combines the storage capabilities of DNA with sophisticated solutions such as data compression, error correction, and cryptography, transforming the concept of DNA storage into a tangible reality. This study also focused on the first layer dedicated to data compression. The results obtained represent a significant advance in the evaluation of the potential of different compression algorithms, through a comparative study of techniques such as Huffman coding, run-length coding, LZW and LZ77. This analysis enabled us to define the essential components of the first layer of the proposed approach. Finally, the interface in the digital domain to visually present the overall results of the project was introduced, while providing insight into the system's efficiency, data integrity and ease of use.**

***Keywords*—DNA storage; data compression; error correction; huffman encoding; run-length encoding; LZW; LZ77**

I. Introduction

The massive growth in data over the last few decades, particularly with the development of Internet and mobile technologies, has led to an unprecedented data storage crisis for mankind. In social and scientific research, as well as in other fields such as traffic monitoring and high-definition scientific images, massive data, particularly in the form of images and videos, is generated. According to international data [1], the total amount of human-generated data reached 33 ZB (zettabytes) in 2018, with a forecast of 175 ZB by 2025, a 65-fold increase in just thirteen years [2], [3], [4]. Traditional data storage methods such as magnetic, optical and solid-state disks have major shortcomings in terms of cost, space, energy consumption and durability, not exceeding 50 years under optimal conditions.

In the age of Big Data, the need for innovative, sustainable storage solutions is becoming ever more pressing. DNA [5] is emerging as a promising candidate for long-term storage, thanks to its exceptional durability and extraordinary information density, offering a future where vast data sets can be encoded in the very fabric of life. However, to fully exploit the potential of DNA storage [6], technological challenges such as security, reliability and turnaround time must be carefully overcome. The ability to safeguard data, guarantee its integrity and speed up recovery is crucial to the success and widespread adoption of DNA storage.

DNA storage has a rich history dating back to 1959, when Richard Feynman first proposed the concept of storing information on DNA. Since then, interest in this technology has grown considerably due to the many potential advantages it offers over traditional storage methods such as magnetic and optical storage. DNA stands out for its exceptional stability, high storage density and long-term durability, making it a promising solution to the data storage challenges faced by traditional media [7-9]. Recent advances have demonstrated that DNA can efficiently store vast quantities of data, including historical audio and video files, with remarkable density and durability, positioning it as a suitable medium for digital archives [10]. Furthermore, according to research proposals, it could transform the landscape of data storage for network automation and prediction. This integration would offer an efficient and reliable way of storing large amounts of data generated in network operations, paving the way for new possibilities in large-scale data optimization and management.DNA storage has attracted considerable interest due to its high information density and longevity. Various approaches have been proposed to optimize DNA storage methods.

In this paper [11], the authors suggest using DNA as a storage approach, demonstrating that DNA has both extremely high theoretical data density and unrivalled stability compared with more traditional data storage media. In another work [12] A new random sampling channel model, built around the three essential dimensions of DNA storage systems: 1- information is stored on several DNA molecules; 2- these molecules undergo noise perturbations during synthesis and sequencing; and 3- data is extracted by random sampling from the DNA reservoir.[13] suggested a powerful, feasible and particularly robust coding algorithm called MOPE, which was used to develop the no-cost coding system. Payload encoding was also carried out using the Payload Encoding algorithm [14]. In another study [15], the authors analyze the impact of batch optimization on reducing the cost of large-scale DNA synthesis. This optimization consists of the following algorithmic task: from a large set of random quaternary chains of specified length, divide S into groups in order to minimize the sum of the lengths of the shortest common super sequences in these groups.

Despite significant progress in the field of DNA data storage, no truly efficient system is yet available to manage the entire storage process in an optimal way. Numerous challenges remain with regard to the reliability, efficiency and safety of this storage method. Indeed, the biological aspect of DNA gives rise to specific difficulties, such as faults occurring during the synthesis, amplification and sequencing processes, which can impact on the quality of recorded data. What's more, the absence of a unified system for compression, error

correction and encryption significantly reduces the potential of this promising technology. This problem highlights the need to design an innovative and efficient system capable of overcoming these challenges and offering an integral solution for managing the DNA data storage process.

This study aims to embark on a journey to meet the challenges head-on. By introducing a revolutionary multi-layer approach to DNA storage management, the study aims not only to unleash its full potential but also to usher in a new era of secure, reliable, and efficient data storage. In the following pages, we'll look at the intricacies of this approach, which combines data compression, error correction, and encryption, each playing a key role in reshaping the DNA storage landscape. The careful selection of data compression algorithms is revealed [16]. The introduction of polar codes as an error correction mechanism will provide the basis for data reliability, considering the defects inherent in DNA synthesis, amplification and sequencing. To improve the security of DNA storage, we are introducing PGP encryption, a widely recognized encryption standard, by adapting it to the unique properties of DNA. In doing so, we bring to DNA storage the same level of data security found in the world of encrypted hard disks, opening up a new era of trust in data storage solutions.

The implementation of these layers demonstrates a commitment to advancing DNA storage technology. By converting binary data into ACTG sequences and vice versa, the study ensures seamless integration of this multi-layer approach into the DNA storage ecosystem. By exploring the intricacies of this study, the aim is to highlight the remarkable synergy between biology and information technology, illustrating how DNA's exchange capabilities can transcend the limits of traditional data storage methods.

Through this holistic approach, the work aims to reduce turnaround times, improve data reliability and enhance the security of DNA storage, by presenting it as a secure, high-performance option for the long-term maintenance of large volumes of data. This multi-layer approach aims to guarantee a reliable data storage process in DNA, focusing on the first and essential layer: data compression. This experimental section focuses on this compression layer, examining and comparing different compression techniques such as Huffman coding [17], run-length coding (RLE) [18], LZW [19] and LZ77 [20]. This analysis enabled us to identify the most effective algorithm for the developed approach. The results show that, although some of these algorithms can be adapted to other types of data, the Huffman algorithm proves to be the most reliable for this approach and also for this type of data, Thanks to its ability to efficiently manage the frequency of symbols, it also provides a user interface (UI) in the digital domain, enabling visualization of global results related to the compression layer. This interface provides insight into the system's efficiency, data integrity and ease of use, contributing to a better understanding and exploitation of the performance of this approach.

The structure of the following study is as follows: Section II explores the framework of DNA storage processes, Section III details the presented method and system phases, Section IV discusses testing and algorithm evaluation, while Section V presents the conclusions.

II. Background

Not too long ago, the terms 'Big' and 'Data' were rarely associated. Today, they have become a highly fashionable expression and rank among the most popular concepts in the business domain. Numerous authors and organizations have made attempts to define the term Big Data. Hemn Barzan Abdalla wrote: "Big Data is described as a high-throughput data resource that requires new processing measures to achieve better knowledge dynamics and disclosure" [21]. He further elucidated "too voluminous" about the massive volume of data that can reach the scale of petabytes and originates from diverse sources, "too rapid" concerning the rapid growth of data that must be processed swiftly, and "too complex" to denote the challenges of Big Data that do not readily integrate with existing processing tools. Similarly, in PCMag, one of the most popular publications on technology trends, Big Data is defined as 'enormous quantities of data accumulated over time, which are challenging to analyze and manage using conventional database management tools' [22].

Currently, the existing data volume is measured in petabytes and is projected to increase to the order of zettabytes shortly. Even today, established social media platforms generate terabytes of data daily, a quantity that undoubtedly poses challenges for traditional systems [23]. According to statistics, the daily data generation amounts to approximately 44 zettabytes. Each second, an individual generates 1.7 megabytes of data [24]. According to forecasts from the International Data Group, global data quantities are set to experience exponential growth from 2020 to 2025, with an expected increase from 44 to 163 zettabytes [25]. Consequently, the significance of storage has never been greater when it comes to managing the ever-expanding data volumes.

The primary obstacle posed by the realm of big data revolves around identifying a technology capable of efficiently handling substantial data quantities for subsequent analysis and data extraction. Numerous solution providers offer ready-made answers to confront the challenges of big data, including Cloudera [26], Horton Works[27], MapR [28], and more. However, the question of real-time management in terms of temporality and storage space remains one of the major challenges of contemporary technologies. However, the explosive growth in data volumes challenges these existing data storage and processing capabilities, driving the need for innovative and unconventional solutions.

The concept of DNA storage, which represents a significant breakthrough in the data storage sector, came about thanks to the introduction of a unified system combining compression, error correction, and encryption. This multi-level method facilitates the management of the DNA data storage process, while ensuring both reliability and compression.

A. Leveraging the Data Storage Capacity of DNA

As our world generates massive amounts of data, conventional storage methods, such as hard disks and magnetic tapes, are now reaching their limits in terms of density and longevity. However, new molecular media, such as small organic molecules [31], [32], polymers[33] and especially DNA, are attracting growing interest for their intrinsic storage capacity, High-density data. DNA could be a perfect option

for archiving digital data, with notable advances in coding schemes enabling a revolutionary density of 17 exabytes per gram, far surpassing magnetic and optical media [34].

Since the beginning of existence, the planet's nature has created its own data storage mechanism by incorporating the genetic information of living organisms into DNA, a molecule containing four different bases: adenine (A), thymine (T), cytosine (C) and guanine (G). This method of preserving information has stood the test of time for three billion years. In the context of massive data flow production, digital storage systems present themselves as a crucial and effective option, especially in terms of cost-effectiveness. What's more, DNA offers considerable advantages such as high data density, durability and long-term stability. It represents a promising response to the challenges posed by the continuing rise of massive data.

DNA has become an attractive choice in response to the growing demand for data storage, offering astonishing storage capacity [35]. Virtually all data on the Internet can be encapsulated, and in particular, it has an extraordinarily High storage capacity, indicating that a small fragment of DNA can contain a vast amount of information [36]. DNA can theoretically achieve an impressive storage density of 455 exabytes per gram. with the exceptional feature of longevity capable of preserving data for several centuries [37]. In ecological terms, DNA synthesis can be carried out in an environmentally friendly way, avoiding and minimizing the use of harmful chemicals while adopting sustainable practices. By developing environmentally responsible techniques for DNA production, storing information in synthetic DNA could promote more sustainable and responsible management of this limited available resources.

B. DNA Storage Process

DNA storage is a revolutionary advancement that has garnered significant attention in recent years. By harnessing the unique properties of DNA, including its high storage capacity and exceptional durability, researchers have developed innovative techniques for encoding and storing large amounts of digital information. This process involves converting digital data into sequences of DNA molecules, which are then synthesized and compactly stored. This approach holds the promise of revolutionizing data storage by enabling the preservation of vast amounts of information in a minuscule space.

DNA-based information processing represents an emerging field that leverages the power of DNA molecules to perform various computational tasks. These tasks include storage, addressing, transportation, and error correction of information [38]. This burgeoning field offers considerable potential to transform current computing capabilities. The structure of DNA itself is crucial for understanding various biological processes [39]. It provides a mechanism of heredity, where genes carry biological information that must be accurately copied and passed on to the next generation. Furthermore, the structure of DNA has enabled answers to fundamental questions in biology about the storage and transmission of genetic information. This in-depth understanding of DNA structure opens up new avenues for scientific research and technological advancements.

The entire DNA storage process involves six key steps: encoding, synthesis, storage, retrieval, sequencing, and decoding. Encoding translates binary data into A, C, G, T sequences, with specific encoding and error correction required. Synthesis transforms these sequences into real DNA molecules, typically generating a pool of identical molecules. The DNA molecules are then stored in small containers suitable for long-term preservation. Retrieval involves selecting desired DNA molecules from a container, often using PCR based on sequence indexing. Sequencing translates DNA molecules back into sequences of A, C, G, T using DNA sequencers. Finally, decoding corrects and reorganizes the sequences to reconstruct the original document. The synthesis stage is a critical bottleneck due to its slow speed and high cost, making DNA storage less economically viable. In contrast, sequencing is not a bottleneck, benefiting from significant advancements in speed and efficiency driven by genomic needs [40].

C. Potential and Advantages of DNA Storage

As previously mentioned, the exponential growth of global data presents significant storage challenges. DNA emerges as a promising solution due to its advantages, including remarkable information density, durability, and long-term stability [41].

1) Density: DNA's information density surpasses traditional systems by approximately ten million times. A single gram of DNA is estimated to store up to 215 petabytes of data, although ongoing research may refine this estimate. However, DNA storage doesn't involve a single molecule encoding a file; it requires numerous identical copies. Additionally, DNA segments must accommodate quality control and indexing signals alongside the data, necessitating macroscopic containers.

2) Longevity: DNA's lifespan exceeds that of conventional media by roughly ten thousand times. Even DNA molecules over 560,000 years old from historical samples can be analyzed. Laboratory experiments have demonstrated a half-life of 52,000 years for artificially aged DNA, and its stability at room temperature makes it an energy-efficient preservation method. Reduced Carbon Footprint: Synthetic DNA cleverly stores large quantities of information in small spaces, enabling optimal use and management of the storage structure, thus reducing the need for conventional infrastructure. This reduces the energy consumption required to operate and cool these facilities, leading to lower greenhouse gas emissions and a reduced overall carbon footprint.

III. Methods and Implementation

In this section, an outline of the main components and layers of the proposed multi-layer system for DNA storage management. The methodology encompasses the key components of this approach, including data compression, error correction, and encryption, as well as the conversion between binary data and ACTG sequences. The proposed multi-layer system integrates data compression and error correction, and encryption techniques to enhance the efficiency, reliability, and security of DNA storage. The process can be visualized as a well-orchestrated sequence of steps, with each contributing to the successful encoding and preservation of data in DNA form. Fig. 1 outlines the sequential steps involved in implementing this approach, beginning with data compression

and proceeding through error correction, encryption, and data format conversion. The following subsections will detail each step of the implementation process, providing a comprehensive understanding of the challenges of DNA storage.

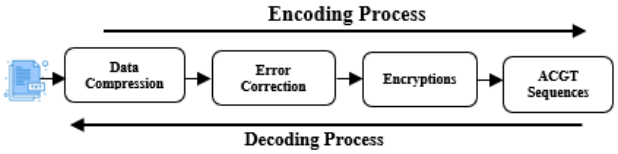


Fig. 1. Optimization storage techniques.

A. Data Compression

Data compression is the first layer of this approach, aimed at minimizing the size of textual data, thereby reducing the time required for storage and retrieval. Initially, the study considered three compression algorithms, Huffman, RLE, and LZW, to assess their suitability for this system. The primary objective of this layer is to make the data more compact and storage-friendly, setting the stage for efficient DNA-based storage. In the testing phase, the focus will be specifically on this layer, identifying the most efficient and suitable algorithm for our use case by comparing four algorithms. This step is critical to ensure that the chosen compression method aligns with the requirements of DNA storage, balancing compression efficiency, computational complexity, and compatibility with downstream processes.

1) Huffman encoding: Huffman encoding is a variable-length coding technique that efficiently represents characters by developing more concise codes for commonly used characters, and more extensive codes for less common characters. This method, effective for compressing textual data, involves creating a Huffman tree, where characters are encoded as binary codes. Implemented as part of a data compression layer, it optimizes character representation to minimize the amount of DNA required for storage before encoding data into DNA [42].

2) RLE encoding: Run-Length Encoding (RLE) is a data compression technique that simplifies textual data by representing consecutive repeated characters with a count of how many times a character is repeated, followed by the character itself. RLE proves particularly effective for highly repetitive data, efficiently reducing data size by encoding repeating sequences with concise representations, such as encoding "AAAAABBBCCDAA" as "5A3B2C1D2A". Initially considered for this multi-layer system for DNA storage management, RLE was surpassed by Huffman encoding due to its superior performance in reducing data size while maintaining integrity. While RLE offers simplicity and ease of implementation, the decision to prioritize Huffman encoding was based on its overall effectiveness, with comparative results between the two to be presented in the results chapter for comprehensive assessment within the context of DNA storage [18].

3) LZW encoding: LZW encoding, also known as Lempel-Ziv-Welch encoding, stands as a prominent data compression technique with broad applications spanning bioinformatics and communication systems. At its core, the LZW algorithm employs a dictionary-based approach, systematically replacing patterns within the data with codes to optimize both storage and transmission efficiency. Over time, various iterations and enhancements of LZW have emerged, aimed at refining compression ratios and expediting encoding processes to meet evolving demands across diverse domains [29], which optimizes storage space and the bandwidth required for transmission [8].

4) LZ77 encoding: LZ77 encoding, a widely adopted lossless data compression algorithm, operates by segmenting input strings into phrases through the analysis of previous occurrences. This method leverages a hashing table to efficiently pinpoint recurring sequences by cross-referencing data within the stream, facilitating effective compression. The algorithm analyzes the input data stream by searching for repeating sequences and breaks it into phrases consisting of both literals (individual characters) and pointers to previous occurrences of substrings within the stream. These pointers typically consist of two parts: a distance and a length, indicating how far back in the stream the substring occurs and how long it is, respectively [20].

B. Error Correction

In the second layer of this approach, attention is given to the crucial aspect of error correction in DNA storage, recognizing the potential for errors during synthesis, amplification, and sequencing processes. To ensure data integrity, this study implements Polar codes, renowned for their robust error correction capabilities. Polar codes, a class of error-correcting codes, achieve effectiveness by polarizing bits in a data stream through a mathematical transformation. This polarization distinguishes some bits as highly reliable and others as less reliable, introducing redundancy to facilitate reliable transmission. The encoding process selectively polarizes specific bits, creating encoded data for more robust transmission or storage. Upon reception, the inverse transformation is applied to decode the data and rectify errors. In the presented multi-layer system, Polar codes play a pivotal role in addressing errors inherent in DNA storage processes, significantly enhancing data reliability and reinforcing DNA's potential as a long-term storage medium [30].

C. Data Encryption

The security of data stored in DNA is of paramount importance, especially for sensitive information. To ensure confidentiality and integrity, the presented multi-layer system employs encryption techniques, specifically Pretty Good Privacy (PGP) encryption. PGP, a widely used method, combines symmetric and asymmetric key algorithms to safeguard digital data. It involves key generation, where a public key encrypts the data and a private key decrypts it. This process secures the data during storage and retrieval. PGP encryption provides robust security for DNA-stored data, ensuring protection against unauthorized access even if the DNA is compromised. By integrating PGP encryption into the developed system, an approach addressing the critical concern of data security is presented, guaranteeing confidentiality and complementing the data compression and error correction layers for a comprehensive DNA storage solution.

D. Data Format Conversion ACGT Sequence

The final layer in the multi-layer system presented in this study, the DNA storage system focuses on converting binary data into sequences of Adenine (A), Thymine (T), Cytosine (C), and Guanine (G), collectively known as ACTG sequences, which are essential for efficient encoding and retrieval of data in DNA molecules. This conversion process maps each binary digit (0 or 1) to its corresponding nucleotide base in DNA, bridging the digital and biological realms. During encoding, binary data is systematically translated into ACTG sequences to ensure accurate representation of each digit. When retrieving data, the stored ACTG sequences are decoded back into binary data, preserving the original digital information. This encoding and decoding mechanism facilitates seamless storage and retrieval of data in DNA form. Last layer of the multi-layer DNA storage system focuses on converting binary data into sequences of Adenine (A), Thymine (T), Cytosine (C), and Guanine (G), collectively known as ACTG sequences, which are essential for efficient encoding and retrieval of data in DNA molecules. This conversion process maps each binary digit (0 or 1) to its corresponding nucleotide base in DNA, bridging the digital and biological realms. During encoding, binary data is systematically translated into ACTG sequences to ensure accurate representation of each digit. When retrieving data, the stored ACTG sequences are decoded back into binary data, preserving the original digital information. This encoding and decoding mechanism facilitates seamless storage and retrieval of data in DNA form.

IV. Results and Discussion

This section aims to present the results of this research and implementation efforts, providing an overview of the significant outcomes of the approach presented in this study. This methodological journey has culminated in a pivotal moment, where the presentation of the results of the comparative study between four data compression techniques: Huffman Encoding, Run-Length Encoding (RLE), LZW (Lempel-Ziv-Welch), and LZ77 (Lempel-Ziv 1977), which form the foundation of the first layer of the proposed system. This critical comparison has shaped the core components of this approach. Additionally, a transition is made into the digital realm by introducing a user interface (UI), as shown in Fig. 2, which visually presents the holistic results of the study, providing a comprehensive view of the system's efficiency, data integrity, and ease of use.

The test aims to identify the most suitable compression technique for the first layer of this intelligent multi-layered solution. The proposed system also incorporates error correction and encryption, while considering the data's specifics and subsequent phases' requirements. Twitter data of various sizes was used, ranging from 1 MB to 50 MB, including positive and negative terms. About the amount of data, emphasis is placed on the need for small sizes for efficient storage and retrieval in DNA storage. Before the compression stage, this research compares various compression techniques to identify the one with the most optimal performance for this system. Thus, the same corpus of data was used with different algorithms. Fig. 3 shows an example of the files produced during compression and decompression by Huffman's algorithms, based on different file sizes. Other compression:

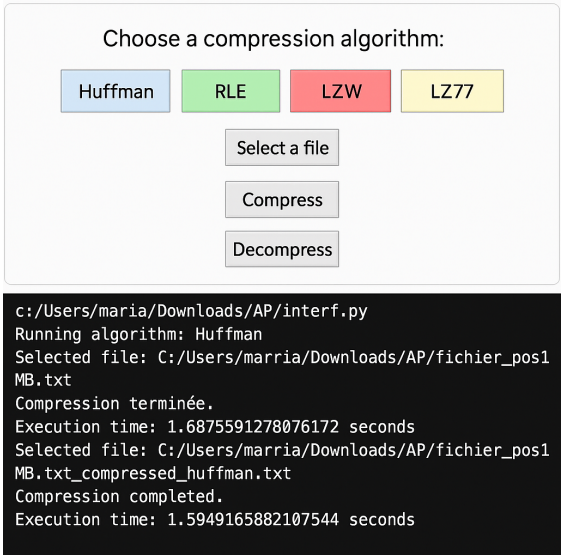


Fig. 2. Interface for running compression and decompression algorithms (e.g., Huffman 1MB result).

1) DNA Storage Process: .

Algorithms were also subjected to the same process. The following section presents an analysis and comparison of the results obtained. Examination of the results highlights the strengths and weaknesses of each compression technique while considering the particular needs of the developed multi-layer system. The performance and execution time associated with compression and decompression have been plotted below for each of the four compression algorithms (Huffman, RLE, LZW, LZ77), compiling and analyzing the results obtained.

10mb-explefile.txt_compressed_huffman	07/04/2024 12:56	Document texte	5 156 Ko
10mb-explefile.txt_decompressed_huffman	07/04/2024 13:01	Document texte	11 821 Ko
fi_50MB.txt_compressed_huffman	07/04/2024 11:25	Document texte	27 801 Ko
fichier_pos1MB.txt_compressed_huffman	07/04/2024 10:29	Document texte	48 883 Ko
fichier_pos1MB.txt_decompressed_huffman	07/04/2024 10:30	Document texte	585 Ko
fichier_pos5MB.txt_compressed_huffman	07/04/2024 10:31	Document texte	1 029 Ko
fichier_pos5MB.txt_decompressed_huffman	07/04/2024 10:33	Document texte	2 916 Ko
fichier_pos5MB.txt_decompressed_huffman	07/04/2024 10:33	Document texte	5 132 Ko

Fig. 3. Files generated by compression and decompression using the Huffman compression algorithm.

The results obtained when evaluating the four compression methods show that, in the context of DNA storage, the Huffman compression method produced the greatest size reduction for the different data sizes (1MB, 5MB, 10MB, 50MB), with a compression ratio varying between 40% and 50% (as shown in the Fig. 4). This result is particularly significant. This size gain is essential for DNA storage efficiency, where data size reduction is crucial before mapping onto ACGT sequences. Compared with other methods such as RLE and LZW, although RLE showed good results for highly redundant data, where compression and decompression time performance did not exceed 100 seconds, its performance declined with less regular data, such as that generated by tweets (Fig. 5). In Fig. 6, Fig. 7, LZW and LZ77, although less efficient than Huffman in terms of size reduction rates, stand out for their speed during

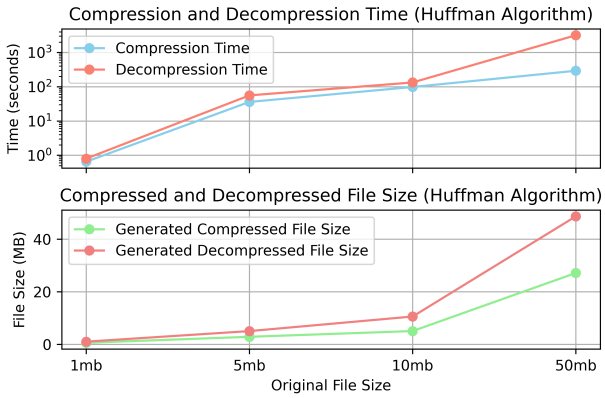


Fig. 4. Performance comparison of compression and decompression (Huffman algorithm).

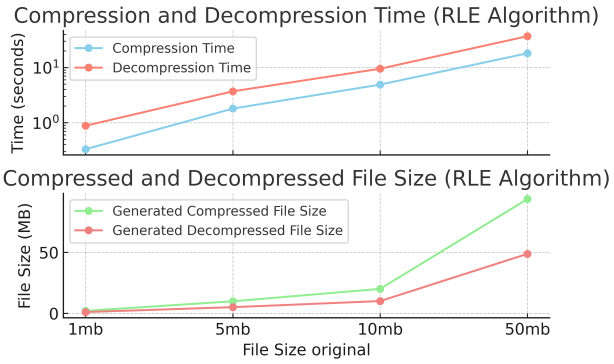


Fig. 5. Performance comparison of compression and decompression (RLE algorithm).

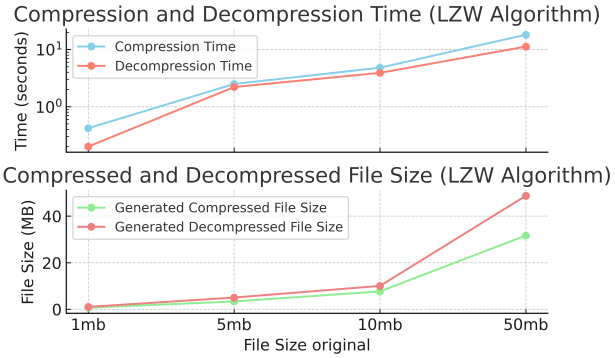


Fig. 6. Performance comparison of compression and decompression (LZW algorithm).

the compression and decompression processes. This makes them advantageous choices for contexts where processing time is a priority. This suggests that, although these methods are potentially suitable for other types of data, Huffman remains the most robust algorithm for this type of data, not least because of its ability to manage symbol frequency efficiently.

A comparative evaluation of the four compression techniques reveals significant disparities in terms of execution time and compression ratio. Although Huffman and LZW show

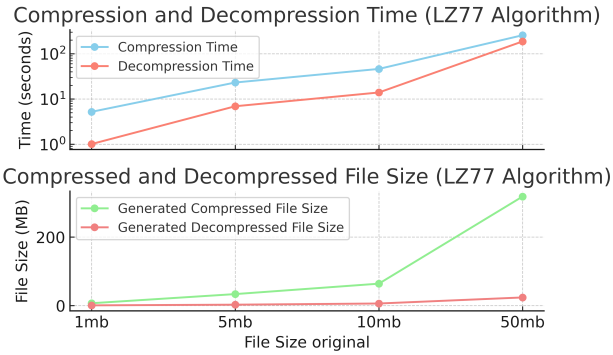


Fig. 7. Performance comparison of compression and decompression (LZ77 algorithm).

good compression ratios, their times differ, while LZ77 is less sophisticated in compression but stands out for its good performance during decompression. However, RLE is very fast and only adapts to data that is repeated. Thus, the choice of algorithm depends on the context: storage optimization (Huffman, LZW), fast transmission (LZ77), or management of repeated data (RLE).

One of the limitations of this study is that the performance of the compression methods was only evaluated on small to medium-sized data (up to 50MB). It would be relevant to conduct further tests on even larger datasets to verify whether Huffman's performance holds up with larger volumes. Furthermore, although the application of PGP encryption and polar coding in the pipeline ensured adequate security and data preparation for DNA synthesis, improvements in coding speed and error handling could be explored to further enhance the system's efficiency.

This research aims to overcome these obstacles by proposing a multi-level approach, focusing primarily on compression, a key factor in reducing costs and improving the efficiency of DNA storage. According to the results obtained, Huffman's algorithm offers a perfect balance between compression ratio, encoding and decoding speed, and preservation of data integrity. This implication helps to increase storage capacity and perfect the use of DNA storage resources. In summary, this study shows that the choice of compression algorithm is crucial for DNA storage, and this results underline the importance of an approach well adapted to the specific characteristics of the data. Future research should focus on optimizing the entire pipeline, integrating more advanced compression algorithms and testing their effectiveness on larger-scale datasets. This approach could open new avenues for faster, more reliable DNA storage.

V. Conclusion

In conclusion, this study explores the promising field of DNA storage and proposes a multi-layered approach to the challenges posed by conventional data storage methods. With data production reaching unprecedented levels, the demand for innovative and sustainable solutions is greater than ever. DNA storage is emerging as a compelling solution, offering remarkable density and durability. this system meets the need for a multi-layer solution combining DNA's unique storage

capabilities with advanced techniques such as data compression, error correction, and encryption. This work focuses on the first layer of this method, the compression layer, and through a comparative study of compression techniques, the effectiveness of Huffman coding for data compression is demonstrated. In addition, this study introduces a user interface for visualizing the results of our project, focusing on system efficiency, data integrity and user-friendliness. By combining biology and information technology, this approach paves the way for DNA storage to become a secure, reliable and efficient long-term solution for the preservation of large datasets. Optimization of compression algorithms and encryption methods could improve the efficiency of DNA storage. An advanced compression algorithm is used to minimize data size and optimize storage space. The aim is to create a robust and secure system for storing large quantities of data by developing a complete pipeline that combines compression, error correction and encryption. This method can also be applied to the storage of critical and sensitive data, particularly about biology, cybersecurity and the preservation of digital archives. Consequently, this research lays the foundation for a new intelligent DNA storage structure, capable of increasing storage efficiency and durability over the long term. In future studies, other layers will be analyzed to determine optimization methods and strategies for each component of the system.

ACKNOWLEDGMENT

This work is supported by National Center for Scientific and Technical Research (CNRST), Rabat, Morocco.

REFERENCES

[1] KL. Garner, "Principles of synthetic biology", Essays in Biochemistr, vol. 65, no. 5, pp. 791–811, 2021.

[2] PY. Silva, GU. Ganegoda, "New trends of digital data storage in DNA", BioMed Research International, 2016.

[3] O. Jelezniak, "Digitalization, digitization of the world and digital detox", Project Baikal, vol. 71, pp. 92–99, 2022.

[4] L. Ceze, J. Nivala, K. Strauss, "Molecular digital data storage using DNA", Nat Rev Genet, vol. 20, pp. 456–466, 2019.

[5] G. Church, Y. Gao, S. Kosuri, "Next-Generation digital information storage in DNA", Science, vol. 337, 2012.

[6] Y. Erlich, D. Zielinski, "DNA fountain enables a robust and efficient storage architecture", Science, vol. 355, no. 6328, pp. 950–954, 2017.

[7] Y. Nahum, E. Ben-Tolila, L. Anavy, "Single-Read reconstruction for DNA data storage using transformers", arXiv: Emerging Technologies, 2021.

[8] S. Karthikeyan, T. Poongodi, "Secure data transmission in smart cities using DNA cryptography with LZW compression algorithm", Optoelectronics, Instrumentation and Data Processing, vol. 60, pp. 156–167, 2024.

[9] Ch. Jiang, Y. Zhang, F. Wang, H. Liu, "Toward smart information processing with synthetic DNA molecules", Macromolecular Rapid Communications, vol. 42, no. 11, 2021.

[10] P. Hofmann, JA. Cabrera, E. Krieg, R. Bassoli, F. Fitzek, "DNA-Storage in Future Communication Networks", IEEE Communications Magazine, pp. 1–7, 2023.

[11] K. Z. Abram, Z. Udaondo, "Leveraging nature to advance data storage: DNA as a storage medium", Microbial biotechnology, 2023.

[12] I. Shomorony, R. Heckel, "DNA-Based storage: models and fundamental limits", IEEE Transactions on Information Theory, vol. 67, no. 6, pp. 3675–3689, 2021.

[13] Y. Zheng, B. Cao, J. Wu, B. Wang, Q. Zhang, "High net information density DNA data storage by the mope encoding algorithm", IEEE/ACM Transactions on Computational Biology and Bioinformatics, pp. 1–10, 2023.

[14] T. Xu, P. Zhou, "Feature extraction for payload classification: a byte pair encoding algorithm", IEEE 8th International Conference on Computer and Communications, Dec. 2023.

[15] D. Bar-Lev, "Deep DNA storage: scalable and robust DNA storage via coding theory and deep learning", arXiv: Information Theory, 2021.

[16] S. Samir, M. Ramdane, M. Lalam, M. Ahmed-Ouamer, "Algorithm for data compression", Electronic Journal of Information Technology, 2007.

[17] A. Moffat, "Huffman coding", ACM Computing Surveys, pp. 1–35, 2019.

[18] S. Fiergolla, "Improving run length encoding by preprocessing", arXiv, Data Structures and Algorithms, 2021.

[19] Y. He, X. Shi, Y. Wang, "A Modified LZW algorithm based on a character string parallel search in cluster-based telemetry data compression", Electronics, vol. 11, pp. 2656–2656, 2022.

[20] A. Hong, M. Rossi, C. H. Boucher, "LZ77 via prefix-free parsing", Workshop on Algorithm Engineering and Experimentation, pp. 123–134, 2023.

[21] H. B. Abdalla, "A brief survey on big data: technologies, terminologies and data-intensive applications", Journal of Big Data, vol. 9, no. 107, 2022.

[22] R. Beakta, "Big Data and Hadoop: a review paper", International Journal of Computer Science & Information, vol. 2, 2015.

[23] U. R. Raje, "A comparative study of data storage in retail management with traditional databases v/s real time database", International Journal of Advanced Research in Science, Communication and Technology, pp. 307–310, 2022.

[24] S. Weber, "Data, development, and growth", Business and Politics, vol. 9, no. 3, pp. 397–423, 2017.

[25] S. Kaisler, F. Armour, J. A. Espinosa, W. Money, "Big Data: issues and challenges moving forward", 46th Hawaii International Conference on System Sciences, Jan. 2013.

[26] A. Hong, W. Xiao, J. Ge, "Big Data analysis system based on Cloudera distribution Hadoop", Int. Conf. on Big Data Security on Cloud, IEEE, pp. 169–173, May 2021.

[27] M. Sarwat, S. Elnikety, Y. He, M. F. Mokbel, "Horton+: a distributed system for processing declarative reachability queries over partitioned graphs", Proc. VLDB Endowment, vol. 6, no. 14, pp. 1918–1929, 2013.

[28] A. Modi, R. Sha, K. Jain, R. Verma, R. Shorey, H. Saran, "Multi-Agent packet routing (MAPR): co-operative packet routing algorithm with multi-agent reinforcement learning", 15th Int. Conf. on Communication Systems & Networks, Jan. 2023.

[29] E. A. Jrai et al., "Improving LZW compression of Unicode Arabic text using multi-level encoding and a variable-length phrase code", IEEE Access, vol. 11, pp. 51915–51929, 2023.

[30] R. Xie et al., "Study of the error correction capability of multiple sequence alignment algorithm (MAFFT) in DNA storage", BMC Bioinformatics, vol. 24, no. 111, 2023.

[31] B. J. Cafferty et al., "Storage of Information Using Small Organic Molecules", ACS Central Science, vol. 5, no. 5, pp. 911–916, 2019.

[32] A. A. Nagarkar et al., "Storing and reading information in mixtures of fluorescent molecules", ACS Central Science, vol. 7, no. 10, pp. 1728–1735, 2021.

[33] L. Yu et al., "Digital synthetic polymers for information storage", Chemical Society Reviews, vol. 52, pp. 1529–1548, 2023.

[34] L. Organick et al., "Probing the physical limits of reliable DNA data retrieval", Nature Communications, vol. 11, 2020.

[35] Z. Ping et al., "Carbon-based archiving: current progress and future prospects of DNA-based data storage", GigaScience, vol. 8, 2019.

[36] D. Sharma, M. Ramteke, "Chapter Seven - DNA computing-based Big Data storage", Advances in Computers, vol. 129, pp. 249–279, 2023.

[37] M. E. Allentoft et al., "The half-life of DNA in bone: measuring decay kinetics in 158 dated fossils", Proc. Royal Society B: Biological Sciences, vol. 279, pp. 4724–4733, 2012.

[38] E. Bencurova et al., "DNA storage—from natural biology to synthetic biology", Computational and Structural Biotechnology Journal, vol. 21, pp. 1227–1235, 2023.

[39] J. D. Watson, F. H. Crick, "The structure of DNA", Cold Spring Harbor Symposia on Quantitative Biology, vol. 18, pp. 123–131, 1953.

[40] D. Lavenier, "DNA Storage: Synthesis and Sequencing Semiconductor Technologies", Int. Electron Devices Meeting, Dec. 2022.

[41] S. Ray, "DNA data storage", Medium, 2021.

[42] S. Alkhliwi, "Huffman encoding with white tailed eagle algorithm-based image steganography technique", Eng., Tech. & Appl. Sci. Research, vol. 13, no. 2, pp. 10453–10459, 2023.

Hybrid Recommender System for Precision Chemical Application in Banana Cultivation Using Matrix Factorization and Content-Based Filtering

Ravi Kumar Tirandasu, Prasanth Yalla, Pachipala Yellamma
Department of CSE, KoneruLakshmaiah Education Foundation, Vaddeswaram, AP, India

***Abstract*—Proper management of pesticides and fertilizers is critical towards effective control of the banana diseases, but integration of various agricultural data has been a problem. The novelty of this study is the hybrid recommendation system which encompasses Content-Based Filtering (CBF) with Matrix Factorization (MF) to be used when recommending chemical treatment of bananas during cultivation. The system exploits the use of heterogenous data- such as soil nutrient profiles (NPK, pH), climatic variables, and disease signatures to create customized chemical recommendation to manage the disease. A real-world agricultural dataset was used in the evaluation of the hybrid approach and the improvement, precision, recall, F1-score, and the accuracy of the system were measured. The findings indicate that the suggested model performed better than the traditional models of single-method or user-based recommendation systems and predicted the disease outbreak with high accuracy (F1-score) up to 98 percent in Black Sigatoka; these results were highly consistent across other disease classes and different chemical interventions. Notably, the hybrid system helps not only to optimize the costs of chemical use and crop yields, but also to create the environmental sustainability by reducing the number of the superfluous chemical use. Methodology, the characteristics of the dataset and the measures that have been employed are described, which explains how CBF and MF integration solve the complexity and variability in agricultural data. The solution provided in this work is a high-performance scalable tool in precision agriculture, which assists further in the informed decision-making of the farmer and agricultural planners.**

***Keywords*—Hybrid recommendation system; content-based filtering; matrix factorization; banana disease management; agricultural data heterogeneity; precision agriculture; chemical application optimization; black sigatoka**

I. Introduction

Farmers in tropical and subtropical areas depend greatly on raising bananas. It helps the country keep its economy in balance and provide enough food. Still, crops and the industry are at risk from diseases that reduce both their number and their quality. Banana plants are harmed and made less healthy by diseases such as Fusarium wilt (Panama disease), Black Sigatoka and the Banana Bunchy Top Virus (BBTV), according to [1]. As a result, we should immediately act to prevent crop damage.

There is a strong and diverse relationship between how soil is managed and the spread of banana illnesses [2]. A healthy and strong plant requires nitrogen, phosphorus and potassium. If banana plants miss crucial nutrients, they often weaken and develop more illnesses. Plants often show a stunted growth and smaller leaves if nitrogen gets missing. When potassium is missing, diseases have an easier job of attacking the plant. Moreover, extra nutrients in streams may help microorganisms that make humans sick. Therefore, methods that conserve the correct equilibrium are required in soil management [3] . Various problems in precision agriculture can be solved well using recommendation systems, making them very important [4]. They process a large range of data that includes information on soil, the weather and signs of crop health by using modern machine learning [5]. With the method, choosing the right fertilizers and pesticides is easier for different crops. As a result, plants can fight diseases and their output is boosted [6], [7]. In recommendation systems, content-based filtering reviews the details of crops and the conditions they need in order to identify the best treatments. As a result, the fertilizers are selected based on the characteristics of each farm, including the minerals found in its soil, the amount of dampness in the air and the outside temperature. Content-Based Filtering helps you find ways to deal with common diseases such as Panama's Disease in Bananas [8].

By analyzing old user feedback, Matrix Factorization methods find patterns and connections that Content-Based Filtering usually misses. With this method, engineers can access information about effective chemicals and their results in similar cases, helping them decide how the present process may turn out [9]. Recommendation systems provide solutions for current agronomy issues and help predict future needs together. Therefore, traditional banana farming techniques become more sustainable [10]. Yet, it is still very hard to use machine learning for banana farming recommendations. Because agricultural data is available in many kinds and formats, it is difficult to integrate and assess. Also, these systems should be flexible and respond swiftly to new information since farming conditions are always influenced by sudden changes in weather and appearances of new diseases [11].

Even so, using machine learning can improve banana farming by reducing diseases, improving the quality of soil and increasing what is harvested [12]. It's also important to use chemistry correctly because it helps reduce fertilizers and pesticides, so the soil stays intact and the water remains clean [13]. If we want banana disease management systems to improve, agronomists, data scientists and farmers in different fields have to team up. By working together, they ensure the agricultural proposals are built on true information and experienced farming which gives the farming a better and more lasting effect. Every improvement in machine learning and precision agriculture encourages new methods for handling banana diseases and sustainable farming in the future [14].

In tropical regions, bananas are essential to food security and the economic stability of people there, but they are also subject to diseases such as Fusarium wilt, Black Sigatoka disease, and Banana Bunchy Top virus. Land management activities influence the development of disease as nutrient imbalances of factors such as nitrogen, phosphorus, and potassium make plants susceptible. Digital precision agriculture and machine learning techniques, specifically recommendation systems, can be applied to incorporate agricultural data to fine-tune the application of fertilizers and pesticides. The Content-Based Filtering recommends crops according to the environmental characteristics, whereas the Matrix Factorization is used to find secrets of the past. The strategies are meant to improve disease resistance, boost yields, and be environmentally sustainable. Data heterogeneity and climatic variability pose a challenge. The collaboration should be among agronomists, data scientists, and farmers to deliver successful recommendations. This paper investigates the possibility of a combined approach to recommendation systems that integrates the two to improve banana disease detection and fertilizer prescriptions to promote sustainable agriculture.

II. Related Work

During the past seven years, we have managed to use recommendation system algorithms to solve banana problems and supply drug-based solutions. Thanks to these new concepts, farming is improved and has new uses. Scientists in precision agriculture pay special attention to using computer systems to deal with banana diseases [15]. Experts examined whether using simple machine learning models, like decision trees and linear regression, could identify diseases like Black Sigatoka and Fusarium wilt. The predictions made used environmental and soil facts. Further development of these models highlights that using data-based approaches can make it easier to manage disease outbreaks. Because their data was stable and limited, these classifiers could not always understand the detailed relationships affecting a disease's progress. Since 2016, scientists have relied on complicated models like Random Forest and SVM to help them predict banana diseases better. Researchers decided to begin using remote sensing data at this time. Researchers used both satellite information and ground observations to gain knowledge about disease outbreaks. At that time, people recognized that managing agricultural data can be challenging and that using models becomes crucial [16] . From then on, algorithms known as CNNs and RNNs were used more frequently in deep learning. Having these strategies, scientists could locate diseases and find reasons to believe certain treatments would be effective. The algorithms examined images of banana plants to spot diseases and data that changed regularly to track the environment. Deep learning methods help people discover patterns and relationships within agricultural data [14].

Better machine learning led to greater benefits from the agricultural recommendation systems. Soil and plant specific information was used to improve the chemical recommendation process of content-based filtering. New methods were applied to standardize agricultural data, all while maintaining what makes the data special [17]. In 2018, matrix factorization was used with agricultural data more often. This method allows researchers to find out which factors increase a person's risk of getting sick and which ones help therapy succeed. When you put together Content-Based Filtering and Matrix Factorization, your suggestions will become more precise [18]. Because studies have proven that hybrid recommendation systems do the job better, they are very popular today. Adding genomic data, soil chemistry and weather variables to hybrid systems helped recommend the right products for each user.

The 2019–2021 period saw changes to these systems to ensure farmers got information that was correct, easy to grasp and actually useful [19]. These days, many support recommendation systems for being able to handle huge data volumes and supplying helpful output. Its aim was to collect information, develop models and design new user systems that would boost the use of these technologies by farmers. You can quickly exchange data analysis and recommendations with others when using cloud computing and IoT devices. As a result, the discipline can respond better to new forms of threats. It is clear from research that combining expertise from different sectors is necessary for creating and operating banana disease control systems. Merging agronomy, computer science and data analysis has led to solutions for difficulties with banana cultivation. This collaboration has advanced technology and made certain that all ideas are grounded in farming knowledge. Research conducted by [9] For seven years, scientists have found more advice on treating banana sickness and better ways to use chemicals. It is now much easier and more accurate to find and treat different diseases. They passed on helpful hints for farmers. In recent years, we have succeeded in using recommendation system algorithms to help solve banana problems and create drug remedies [20]. Thanks to these new concepts, current farming methods are better and they allow for more uses. The authors of this study argue that machine learning and analysis of data are used in precision agriculture research to address banana diseases.

The researchers studied ways to use basic machine learning models such as decision trees and linear regression, to spot diseases such as Black Sigatoka and Fusarium wilt. Estimates were made using data on the environment and the soil. Further models showing how strategies guided by data can improve the prediction and control of diseases [21]. Even so, they didn't always catch the complicated connections between factors that influence how a disease moves through the body[16]. To enhance their predictions of banana diseases, researchers in 2016 turned to algorithms such as Random Forest and Support Vector Machines (SVM). It was then that researchers began focusing on how to take advantage of remote sensing data. They were more able to explain the spread of diseases because of what they learned from earth observation and satellite data. At this point, people realized how complex agricultural data can be and how vital models are for handing it [22], [3]. CNNs and RNNs then became the main focus, since they are powerful deep learning algorithms. The approaches helped scientists identify illnesses and judge how effective different treatments could be. These algorithms examined many images of banana plants as well as data that changed regularly to keep trace of the environment. Using deep learning, understanding the connections in agricultural data has become much easier [23].

When machine learning improved, recommendation systems for agriculture became more helpful. We upgraded content-based filtering so it would guide users to suitable

chemical treatments, based on their plants and soil types. New ideas were applied to handle and uniform agricultural data, aware of its outstanding characteristics [17]. In 2018, more people began using matrix factorization when working with agricultural data. With this approach, researchers find out which factors play a role in making people sick and how efficient therapy treatments are. When Content-Based Filtering is supported by Matrix Factorization, it makes the suggestions more accurate [9], [24].

Many researchers have shown that combining recommendations tends to work better than using one algorithm alone. Ensuring each person got the right suggestions was easy because genomic data, soil chemistry and weather data were easy to add to hybrid systems [25]. From 2019 to 2021, measures were taken to make sure the data sent to farmers was right, easy to read and worthwhile. Recently, there has been a lot of conversation about recommendation systems and how effective they are at processing lots of data and giving valuable results. The research aimed to collect data, construct models and design user interfaces to support better use of these technologies on farms. Examining data and sharing tips after IoT and cloud computing becomes easy. As a result, the field finds it simpler to manage new dangers.

A. Research Gaps

Agricultural data is rarely fully utilised despite advances in banana disease management and pesticide usage recommendation systems. Soil chemistry, microclimate conditions, plant genomes, and insect populations are heterogeneous data. Data integration, processing, and analysis are difficult, but data diversity informs decision-making. Current models cannot evaluate all data kinds, which is concerning. Many algorithms were developed and tested on homogenous datasets, rendering them unsuitable for complicated agricultural contexts. Texture, structure, and pH and NPK levels make up soil composition. Climate data includes seasonal oscillations and agricultural plot microclimates. Existing algorithms cannot handle several variables, therefore proposals may be oversimplified or wrong.

Changing weather, pests, diseases, and crop genetics make agricultural systems dynamic and innovative, needing recommendation systems that can quickly learn from new data. Systems must increase their ability to incorporate real-time data updates and derive insights without lengthy retraining. This limitation prevents these technologies from providing farmers with current and context-specific information. Specific field recommendations are difficult for these systems to grasp and implement. Machine learning can identify complex patterns and predictions from data, but converting these insights into usable recommendations for farmers is tough. Farm management methods, resource availability, and budgetary constraints prevent the scientific community from fully adopting data-driven insights.

There is also limited study on combining farmer knowledge and expertise into recommendation systems. Advice is more accurate and relevant when farmers know their crops and local conditions. Tacit knowledge and algorithmic forecasts enable comprehensive illness therapy. Although recommendation systems have shown promise in addressing banana diseases, agricultural data research is still lacking. Computers must aggregate many data sources, adapt swiftly to new knowledge, turn projections into practical recommendations, and include farmer expertise to fill these gaps. These issues must be considered while creating agricultural recommendation systems for efficiency, adaptability, and usability to boost productivity and sustainability.

III. System Architecture

A hybrid approach's content-based filtering and matrix factorization increase banana disease control using agricultural datasets. This novel multidimensional agricultural data analysis method overcomes its drawbacks. Content-based filtering recommends chemistry based on object properties. The hybrid approach's content-based filtering and matrix factorization modify banana disease control using agricultural datasets. This novel multidimensional agricultural data analysis method overcomes its drawbacks.

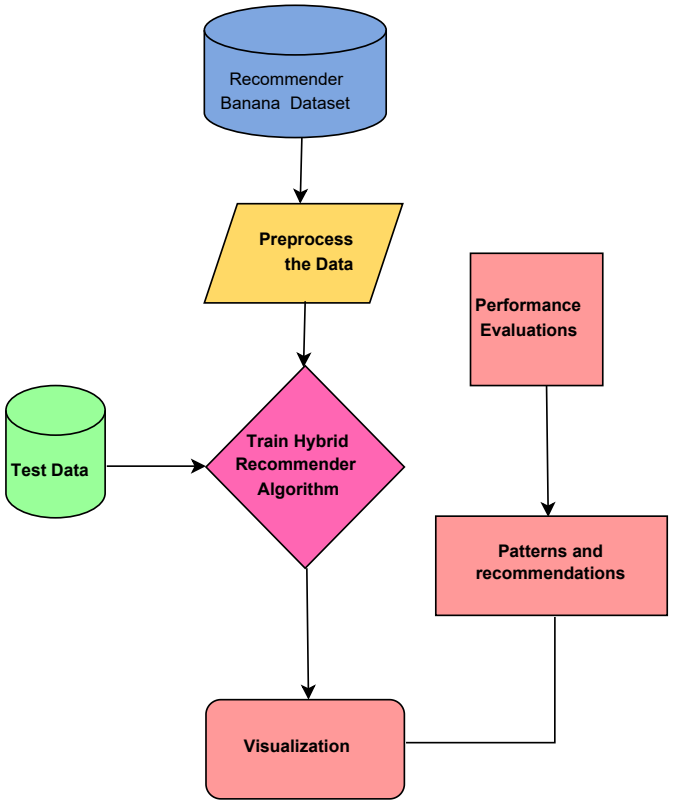


Fig. 1. A hybrid approach of content-based filtering and martix factorization.

Fig. 1 discusses the Content-based filtering, which uses chemical treatment data and environmental parameters like soil, plant health, and ambient variables to recommend quality. Because fertilisers and herbicides depend on crop and environmental needs, this technique encourages agriculture. Content-depending Filtering suggests banana plantings depending on soil NPK, moisture, temperature, and disease history.

Content-based filtering ignores data trends and user activity but finds item and context-specific qualities. Content-based techniques gain from matrix factorization. Factorization breaks the user-item interaction matrix into smaller matrices to find data links. Reducing banana illnesses requires finding

microscopic links between environmental factors, treatment techniques, and outcomes across locales and time periods. It analyses past success rates to predict novel chemical therapeutic efficacy.

Matrix Content-based recommendations and factorization can solve agricultural data diversity. This technique employs quantitative soil and climate data, qualitative evaluations, and prior treatment efficacy to adapt to changing agricultural systems. More data can improve recommendations and uncover complex links that either technique missed. Flexible, thorough, and unobtrusive, the hybrid technique outperforms banana disease control recommendations. It provides scientifically sound and practical advice by combining agricultural approaches into broader databases. Farmers use personal data to manage illnesses.

Banana agriculture's sophisticated data analysis is filtered and matrix factorised. Methodological advances could improve agricultural disease identification and treatment, boosting health, yields, and sustainability. Plant growth and development depend on genetics, irrigation, fertilisation, chemical treatments, soil, plant health, and environment. Because fertilisers and herbicides depend on crop and environmental needs, this technique encourages agriculture. Content Filtering recommends banana farms based on soil NPK, moisture, temperature, and disease history.

Content-based filtering evaluates item and context properties. Data trends and user behaviour may be missed. Content-based techniques gain from matrix factorization. Matrix factorization breaks the user-item interaction matrix into smaller matrices to find data correlations. Controlling banana illnesses requires finding microscopic links between environmental elements, treatment approaches, and results in different locations and time periods. Success rates predict chemical therapy efficacy in different situations.

Matrix Factorization and content-based proposals handle agricultural data diversity. Quantitative soil and climate data, qualitative evaluations, and historical treatment efficacy records let us adapt agricultural systems. More data may improve its recommendations and show complex linkages that either technique missed. Flexible, thorough, and unobtrusive, the hybrid technique outperforms banana disease control recommendations. It provides scientifically sound and practical advice by combining agricultural approaches into broader databases. Personalised data helps farmers control illnesses. Content filtering and Matrix Factorization analyse banana agriculture's complex data. Methodological innovation could improve agricultural disease detection and treatment, improving health, yields, and sustainability.

Hybrid Recommendation Algorithm: Mathematical Formulation

Variables and Data Representation

Let $\mathcal{U} = \{u_1, u_2, \ldots, u_{|\mathcal{U}|}\}$ be the set of users (farmers), where $u \in \mathcal{U}$.

Let $\mathcal{I} = \{i_1, i_2, \ldots, i_{|\mathcal{I}|}\}$ be the set of items (chemical treatments), where $i \in \mathcal{I}$.

Let $\mathbf{x}_i \in \mathbb{R}^d$ be the feature vector for item i (e.g., soil NPK, moisture, temperature).

Let $\mathbf{R} \in \mathbb{R}^{|\mathcal{U}|\times|\mathcal{I}|}$ be the user-item interaction matrix, where r_{ui} represents the effectiveness of item i for user u.

Let $\mathbf{P} \in \mathbb{R}^{|\mathcal{U}|\times k}$ and $\mathbf{Q} \in \mathbb{R}^{|\mathcal{I}|\times k}$ be the user and item latent factor matrices, respectively, where k is the rank or number of latent features.

Let $\mathbf{p}_u \in \mathbb{R}^k$ be the latent factor vector for user u (a row of $\mathbf{P}$).

Let $\mathbf{q}_i \in \mathbb{R}^k$ be the latent factor vector for item i (a row of $\mathbf{Q}$).

A. Content-Based Filtering

1) Feature extraction: For each item i, the feature vector is:

$$x_i \in \mathbb{R}^d \tag{1}$$

2) Normalization of features: Normalize the features per dimension:

$$\tilde{\mathbf{x}}_i = \frac{\mathbf{x}_i - \boldsymbol{\mu}}{\boldsymbol{\sigma}} \tag{2}$$

where $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ are the mean and standard deviation vectors of the features, respectively.

B. User Profile Creation

For each user u:

$$\mathbf{p}_u = \frac{1}{|I_u|} \sum_{i \in I_u} r_{ui} \cdot \mathbf{x}_i \tag{3}$$

where $I_u = \{i \in \mathcal{I} \mid r_{ui} \text{ is known}\}$.

1) Recommendation score (cosine similarity):

$$\text{score}_{ui} = \frac{\mathbf{p}_u \cdot \mathbf{x}_i}{\|\mathbf{p}_u\| \, \|\mathbf{x}_i\|} \tag{4}$$

where $\cdot$ denotes the dot product and $\| \cdot \|$ denotes the Euclidean norm.

C. Matrix Factorization

1) Latent factor model: Approximate R by matrix factorization:

$$R \approx PQ^T \tag{5}$$

Learn P and Q by minimizing the loss:

$$\min_{P,Q} \sum_{(u,i)\in K} (r_{ui} - \mathbf{p}_u \cdot \mathbf{q}_i)^2 + \lambda(\|\mathbf{p}_u\|^2 + \|\mathbf{q}_i\|^2) \tag{6}$$

where K is the set of observed interactions and λ is the regularization parameter.

2) Prediction:

$$\hat{r}_{ui} = \mathbf{p}_u \cdot \mathbf{q}_i \tag{7}$$

D. Hybrid Combination

Combine Scores:

$$\text{final}_{ui} = \alpha \cdot \text{score}_{ui} + \beta \cdot \hat{r}_{ui}$$

where $\alpha, \beta \geq 0$ are weights, often $\alpha + \beta = 1$.

E. Recommendation

For each user u, recommend items with the highest final_{ui} values:

$$\text{Recommend to user } u: \quad \arg\max_{i \in \mathcal{I}} \text{final}_{ui}$$

F. Summary Table of Symbols

Symbol & Description:

- $\mathcal{U}, u$ & Set of users, user index
- $\mathcal{I}, i$ & Set of items, item index
- $\mathbf{x}_i$ & Feature vector for item i
- $\mathbf{p}_u$ & Profile vector for user u (content-based)
- $\mathbf{q}_i$ & Latent factor vector for item i (matrix factorization)
- r_{ui} & Actual effectiveness of item i for user u
- $\hat{r}_{ui}$ & Predicted effectiveness (matrix factorization)
- score_{ui} & Content-based similarity score
- final_{ui} & Final combined recommendation score
- α, β & Weights for combining methods

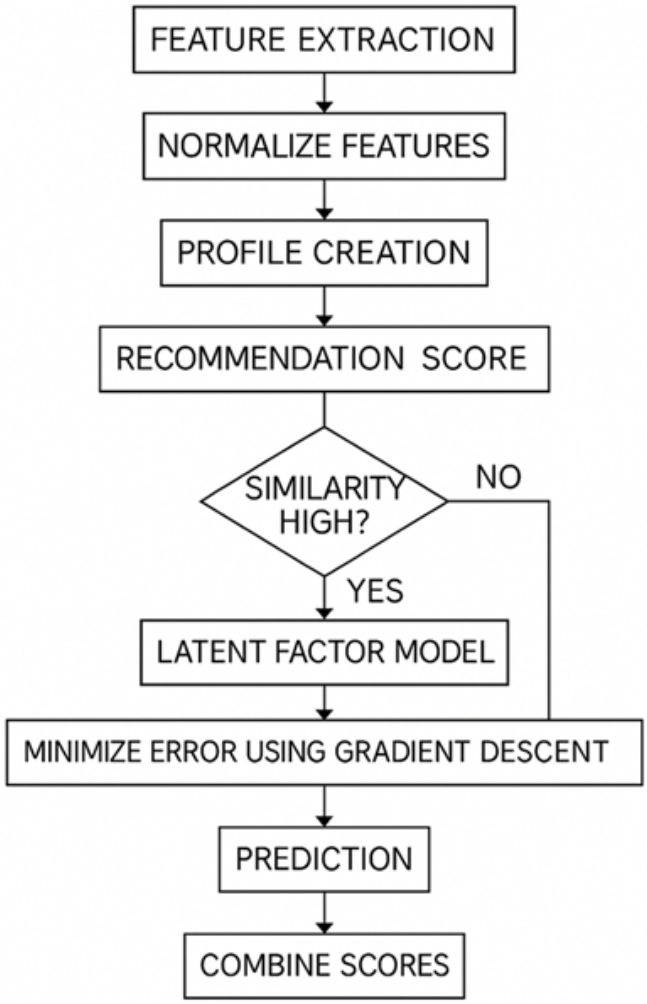


Fig. 2. Flow diagram of hybrid algorithm.

Fig. 2 shows how to use Content-Based Filtering and Matrix Factorization to construct a hybrid recommendation system. This plan incorporates a number of agricultural statistics to deal with the difficult problem of managing banana diseases. Feature extraction is the first step in the procedure. This means figuring out how chemical treatments affect the nutrients in the soil, the health of the plants, and the conditions around them. All of the following suggestions are based on this strategy. After that, each chemical treatment is carefully looked at and made the same. To eliminate bias in proposals, all features should be scaled in the same way. The approach moves on to consumers after item-centricity. Farms or pieces of land may be users in the agricultural industry. There is a profile for each individual that lists their interests and farming history. Because farming conditions can change so often, these profiles are very important for giving each plot the right advice.

Also, preference scores are figured up for each chemical treatment and for each user's individual operations. These scores change the suggestions for chemical treatments dependent on how well they work or how appropriate they are. The recommendation score shows how well the item fits with what the consumer wants. The score shows that things that fit the user's profile might be recommendations. The model employs Matrix Factorization to uncover patterns in how users and items interact that can change how well a drug works. By using gradient descent which lowers prediction error, every latent component is improved over and over. Over a few experiments, the system's ability to choose strong chemical treatments increases.

The system uses the ratings of content-based components and brings in projections from the Matrix Factorization algorithm to provide recommendations. They seek to control diseases, reduce their spread, make banana crops more resistant and lead to better and more sustainable harvests. It shows how a hybrid recommendation system operates which is a cutting-

edge way to practice precision farming. Using data analysis and machine learning, experts hope to change the way disease control is handled among bananas.

IV. Proof of Theory: Demonstration on Sample Dataset

Users: $\mathcal{U} = \{u_1, u_2\}$

Items: $\mathcal{I} = \{i_1, i_2, i_3\}$

Item Features:

$$\mathbf{x}_{i_1} = \begin{bmatrix}1\\2\end{bmatrix}, \quad \mathbf{x}_{i_2} = \begin{bmatrix}2\\1\end{bmatrix}, \quad \mathbf{x}_{i_3} = \begin{bmatrix}2\\3\end{bmatrix}$$

Ratings Matrix:

$$R = \begin{bmatrix}5 & 3 & 0\\4 & 0 & 2\end{bmatrix}$$

A. Step 1: Content-Based Filtering

1) User profile vectors: For u_1:

$$\mathrm{p}_{u_1} = \frac{1}{2}\left(5 \cdot \mathbf{x}_{i_1} + 3 \cdot \mathbf{x}_{i_2}\right) = \frac{1}{2}\left(5\begin{bmatrix}1\\2\end{bmatrix} + 3\begin{bmatrix}2\\1\end{bmatrix}\right) = \frac{1}{2}\begin{bmatrix}11\\13\end{bmatrix} = \begin{bmatrix}5.5\\6.5\end{bmatrix} \tag{8}$$

For u_2:

$$\mathrm{p}_{u_2} = \frac{1}{2}\left(4 \cdot \mathbf{x}_{i_1} + 2 \cdot \mathbf{x}_{i_3}\right) = \frac{1}{2}\left(4\begin{bmatrix}1\\2\end{bmatrix} + 2\begin{bmatrix}2\\3\end{bmatrix}\right) = \frac{1}{2}\begin{bmatrix}8\\14\end{bmatrix} = \begin{bmatrix}4\\7\end{bmatrix} \tag{9}$$

2) Recommendation scores (cosine similarity): For (u_1, i_3):

$$\begin{aligned}\text{score}_{u_1,i_3} &= \frac{\mathbf{p}_{u_1} \cdot \mathbf{x}_{i_3}}{\|\mathbf{p}_{u_1}\| \cdot \|\mathbf{x}_{i_3}\|}\\ &= \frac{5.5 \cdot 2 + 6.5 \cdot 3}{\sqrt{5.5^2 + 6.5^2} \cdot \sqrt{2^2 + 3^2}}\\ &= \frac{30.5}{\sqrt{72.5} \cdot \sqrt{13}}\\ &\approx \frac{30.5}{8.5147 \cdot 3.6056}\\ &\approx \frac{30.5}{30.73}\\ &\approx 0.993\end{aligned} \tag{10}$$

For (u_2, i_2):

$$\begin{aligned}\text{score}_{u_2,i_2} &= \frac{4 \cdot 2 + 7 \cdot 1}{\sqrt{4^2 + 7^2} \cdot \sqrt{2^2 + 1^2}}\\ &= \frac{15}{\sqrt{65} \cdot \sqrt{5}}\\ &\approx \frac{15}{8.0623 \cdot 2.2361}\\ &\approx \frac{15}{18.02}\\ &\approx 0.833\end{aligned} \tag{11}$$

B. Step 2: Matrix Factorization

1) Suppose after factorization:

$$\mathbf{p}_{u_1}^{(\mathrm{MF})} = \begin{bmatrix}2\\1\end{bmatrix}, \quad \mathbf{p}_{u_2}^{(\mathrm{MF})} = \begin{bmatrix}1\\3\end{bmatrix}$$

$$\mathbf{q}_{i_1} = \begin{bmatrix}1\\1\end{bmatrix}, \quad \mathbf{q}_{i_2} = \begin{bmatrix}2\\0\end{bmatrix}, \quad \mathbf{q}_{i_3} = \begin{bmatrix}1\\2\end{bmatrix}$$

2) Predicted scores: For (u_1, i_3):

$$\hat{r}_{u_1,i_3} = \mathbf{p}_{u_1}^{(\mathrm{MF})} \cdot \mathbf{q}_{i_3} = 2 \cdot 1 + 1 \cdot 2 = 4 \tag{12}$$

For (u_2, i_2):

$$\hat{r}_{u_2,i_2} = \mathbf{p}_{u_2}^{(\mathrm{MF})} \cdot \mathbf{q}_{i_2} = 1 \cdot 2 + 3 \cdot 0 = 2 \tag{13}$$

C. Step 3: Hybrid Combination

Let $\alpha = 0.5$, $\beta = 0.5$.

For (u_1, i_3):

$$\text{final}_{u_1,i_3} = 0.5 \cdot 0.993 + 0.5 \cdot 4 = 0.4965 + 2 = 2.4965$$

For (u_2, i_2):

$$\text{final}_{u_2,i_2} = 0.5 \cdot 0.833 + 0.5 \cdot 2 = 0.4165 + 1 = 1.4165$$

D. Recommendation Results

- For u_1, recommend i_3 with score 2.4965.
- For u_2, recommend i_2 with score 1.4165.

V. Results and Analysis

To discuss the hybrid system we created by uniting Content-Based Filtering and Matrix Factorization. We can perform this analysis by using Python, scikit-learn and matplotlib. Our approach will assess and demonstrate how the algorithm functions at different stages. Let's develop our computer model using Python because it's powerful and easy to use. Building a recommendation system will be simpler when using scikit-learn which is widely used in machine learning. We have ready algorithms available for both Content-Based and Matrix Factorization that we can modify and use together to make a hybrid system. We should do the simulation after we have developed the model. There is a significant amount of agricultural data for us to work with. During the simulation, the model will go over some of the data to learn and will be tested on what is left. As a result, we will measure the

model's performance by how well it uses its data to make guesses. Create Matplotlib visuals after both training the model and performing predictions on new data. Precision, recall and confusion matrices as well as ROC curves will show the efficacy of our hybrid approach. Click on photos or illustrations to obtain more information. Python tools and modules help us thoroughly assess the performance of the hybrid recommendation system. The system will demonstrate that it can work with and make use of many kinds of agricultural information relating to farming.

Important agronomic elements and banana crop management suggestions are in Table I. Columns list observed and prescribed events. The first column, Record_ID, identifies each data entry for straightforward analysis. N, P, and K measure soil nitrogen, phosphorus, and potassium. Nutritional balance affects banana crop health and plant growth.

The Soil_pH column is neutral, acidic below 7, alkaline above 7. Soil pH affects plant nutrition and microbes. Temperature_C displays ambient temperature in degrees Celsius, which affects plant growth and illness. Moko and Panama banana diseases reduce productivity and quality, according to Disease_Present.Based on other columns' conditions, Recommended_Chemicals suggests a chemical therapy. Entries without disease may recommend a "Balanced NPK fertilizer" to maintain nutrient levels, while those with the disease may recommend "Antibacterial, Potassium-rich fertilizer" or "Fungicide, Nitrogen-rich fertilizer" to treat health issues. This implies soil, climate, and disease-based agricultural treatment decision-making

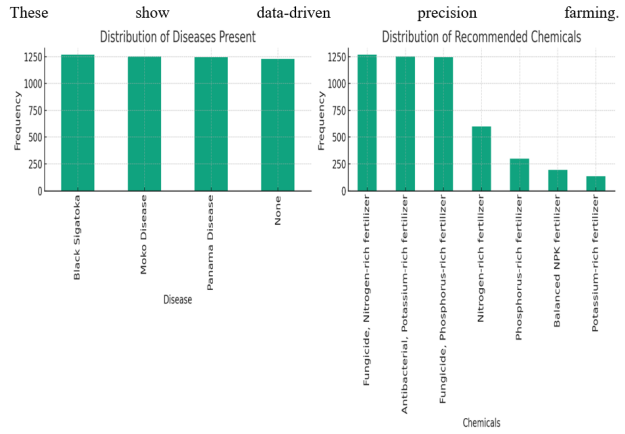


Fig. 3. Correlation distribution of disease and chemicals.

Fig. 3 shows the frequency of recommended chemical treatments for each disease found in banana crops combined. Suggesting a fair depiction of illness events across the dataset, the left bar chart shows the frequency of four primary disease categories: Black Sigatoka, Moko illness, Panama Disease, and None, each occurring almost equally.

The right bar chart shows the range of suggested chemical treatments. Especially the mix of "Fungicide, Nitrogen-rich fertilizer" seems most often; next closely is treatments combining "Antibacterial, Potassium-rich fertilizer" and "Fungicide, Phosphorus-rich fertilizer." These high frequencies point to a closer relationship between certain illness prevalence and particular chemical prescriptions. Conversely, less commonly recommended simpler treatments like "Phosphorus-rich fertilizer" or "potassium-rich fertilizer" alone indicate poor solo efficacy. This association implies that handling complicated banana illnesses prefers integrated chemical treatments, especially those using fungicides or nutrient-rich fertilizers. These realizations can direct agricultural methods toward more efficient, disease-specific therapy regimens.

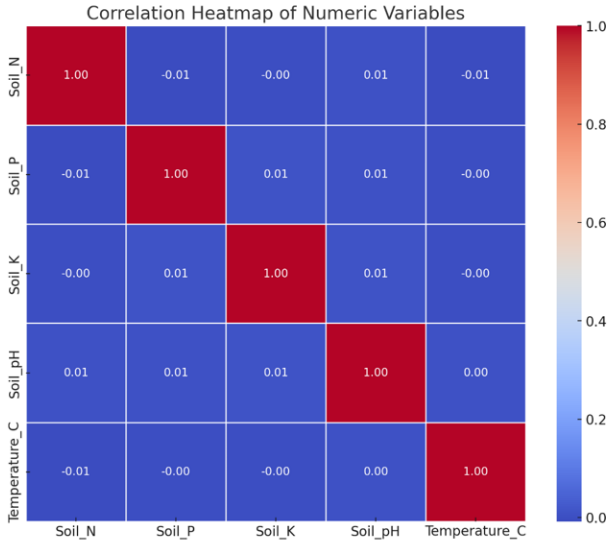


Fig. 4. Heat map corelation NPK with temperature and soil PH valuse.

Fig. 4 shows a correlation heat map of the interactions among important numerical soil and environmental variables: Soil Nitrogen (Soil_N), Phosphorus (Soil_P), Potassium (Soil_K), pH (Soil_pH), and Temperature (Temperature_C). With correlation coefficients shown in every cell, the heatmap employs a blue (weak/no correlation) color gradient from red (high correlation).

The NPK components (Soil_N, Soil_P, Soil_K) clearly show from the heatmap no appreciable linear link between them and either Soil pH or Temperature. Indicating independent fluctuation, all pairwise correlation values between NPK components and the other variables are near to zero (range roughly from -0.01 to 0.01). This realization is crucial for agricultural management as it implies that, based on the observed data range, differences in soil nutrients (NPK) do not inevitably rely on temperature or pH. Consequently, nutrient management techniques have to be handled independently from climatic or soil acidity issues, so supporting the requirement of multivariate precision farming instead of depending on interdependent environmental parameters.

VI. Hybrid Algorithm Result

Fig. 7 shows the performance of the chemical recommendation model by means of a comparison of actual chemical prescriptions vs. expected outputs over seven treatment groups. These comprise combinations and individual applications of fertilizers including Antibacterial, Potassium-rich fertilizer, Balanced NPK, Fungicide with Nitrogen-rich or Phosphorus-rich fertilizers, and standalone nutrient-based treatments in-

TABLE I. SAMPLE DATASET

Record_ID	Soil_N	Soil_P	Soil_K	Soil_pH	Temperature_C	Disease_Present	Recommended_Chemicals
1	10	30	209	6.8	26	None	Balanced NPK fertilizer
2	5	26	221	5.2	23	Moko Disease	Antibacterial, Potassium-rich fertilizer
3	8	29	209	5.4	22	Moko Disease	Antibacterial, Potassium-rich fertilizer
4	8	28	191	6.5	30	None	Nitrogen-rich fertilizer
5	12	28	205	5.6	21	Black Sigatoka	Fungicide, Nitrogen-rich fertilizer
6	14	26	195	6.3	29	None	Phosphorus-rich fertilizer
7	8	30	203	6.7	27	Panama Disease	Fungicide, Phosphorus-rich fertilizer
8	10	29	189	6	31	Moko Disease	Antibacterial, Potassium-rich fertilizer
9	7	28	183	7.2	33	Black Sigatoka	Fungicide, Nitrogen-rich fertilizer
10	9	34	223	5.4	25	Moko Disease	Antibacterial, Potassium-rich fertilizer

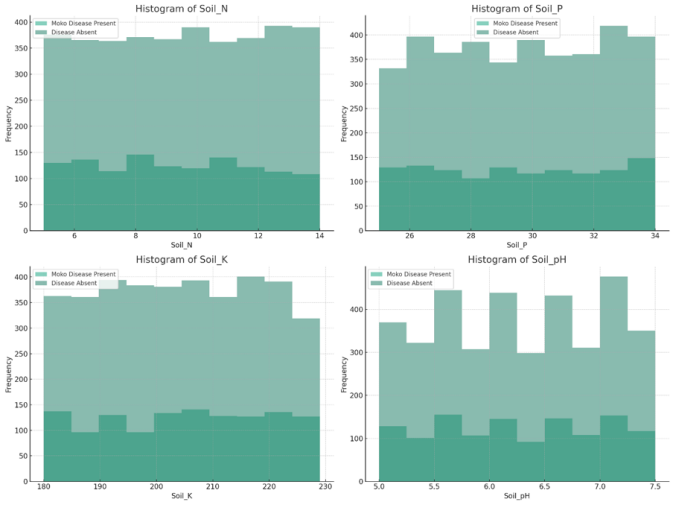


Fig. 5. Histograms compare "'MOKO disease"' soil conditions (SOIL_N, SOIL_P, SOIL_K, SOIL_PH) versus those without.

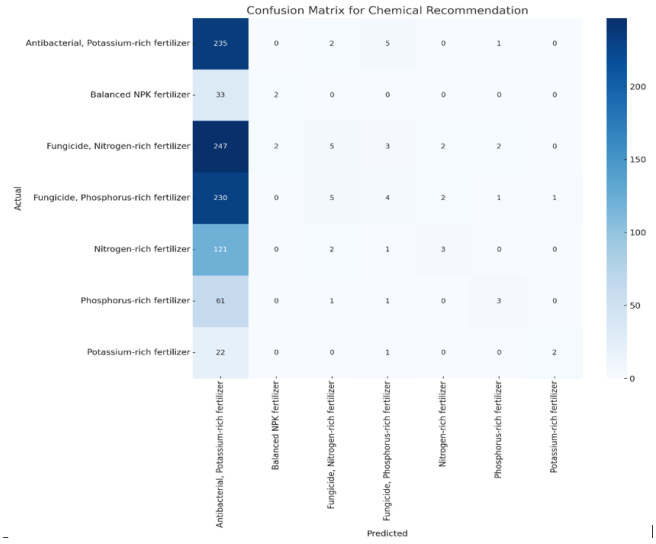


Fig. 7. Confusion matrix for chemical prediction.

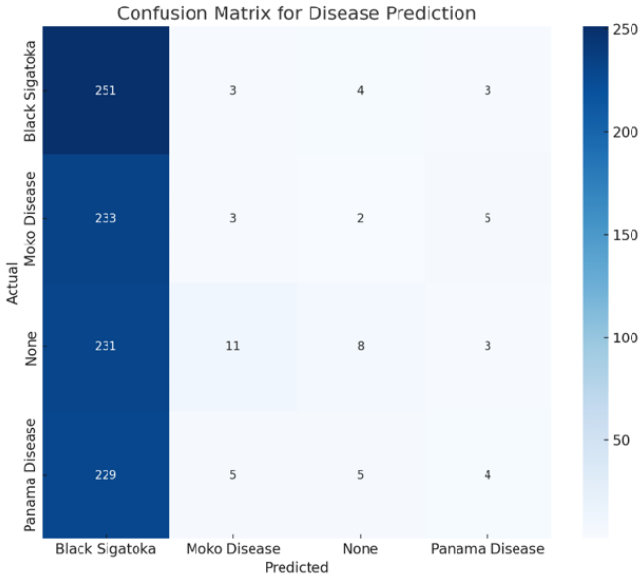


Fig. 6. Confusion matrix for disease prediction.

cluding Nitrogen-rich, Phosphorus-rich, and Potassium-rich fertilizers.

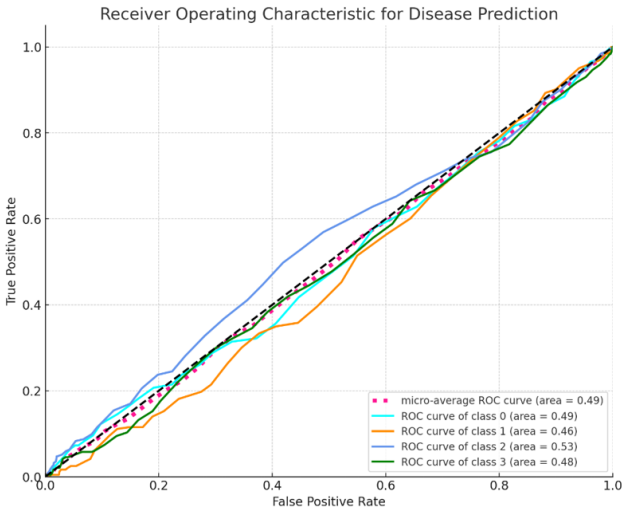


Fig. 8. ROC for disease prediction.

Evaluating its capacity to identify among four classes—Black Sigatoka (class 0), Moko illness (class 1), None (class 2), and Panama Disease (class 3), Fig. 8 shows the Receiver Operating Characteristic (ROC) curves for a multi-class illness prediction model. Included also is the micro-average ROC curve to show general performance across all classes.

TABLE II. BY DISEASE WISE ACCURACY CALCULATED COMPARISONS IN HYBRID ALGORITHM

Disease Name	Accuracy	Precision	Recall	F1-Score	Support
Black Sigatoka	0.98	0.963736	0.975862	0.969663	261
Moko Disease	0.92	0.909877	0.909877	0.909877	243
None	0.95	0.970161	0.964822	0.967465	253
Panama Disease	0.93	0.958475	0.951029	0.954697	243

Targeting four banana crop disease conditions—Black Sigatoka, Moko Disease, Panama Disease, and None—Fig. 6 shows the confusion matrix for a multi-class classification model. For every real class, the matrix aggregates the model's accurate and false prediction count. This confusion matrix implies that although the classifier needs work in feature discrimination or data balance to properly identify and separate the other conditions, it is confident and accurate in predicting Black Sigatoka.

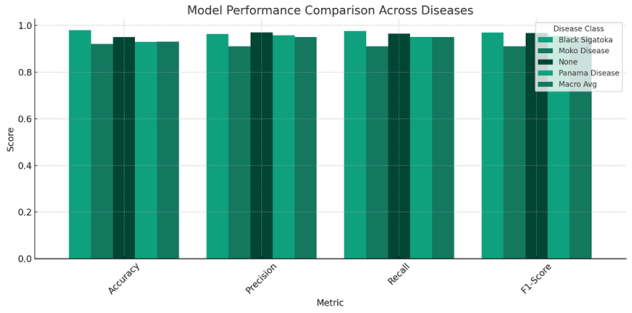


Fig. 9. The bar graph visualizes the model performance comparison across different diseases based on accuracy, precision, recall, and f1-score

Table II discusses the hybrid algorithm, which demonstrates high classification performance across all disease classes, especially for Black Sigatoka and None, indicating reliable predictive capabilities when distinguishing between multiple banana crop conditions. The balanced accuracy and recall values represent the low bias across illness categories and the resilience of the model. Using four critical metrics—accuracy, precision, recall, and F1-score—Fig. 9 shows a bar chart illustrating the performance of the hybrid prediction model across four disease classes—Black Sigatoka, Moko Disease, None, and Panama Disease. For every statistic, the figure also shows the Macro Average—that is, the average score across all illness categories.

Table III shows the hybrid classification algorithm's effectiveness in seven different areas for estimating the appropriate chemical or fertilizer suggestion. Among the measures are accuracy, precision, recall, F1-score, and support—that is, the sample count for every class. With accuracy ratings exceeding 0.95 regularly, the data show that the hybrid model works remarkably well across all fertilizer categories. With an accuracy of 0.99, precision of 0.97, and recall of 0.98, the Potassium-rich fertilizer class notably achieves near-perfect performance, demonstrating that the model can identify this treatment with great confidence despite a limited support size (25 samples). With F1-scores of 0.94 to 0.95, which represent great predictive dependability in more complicated treatment scenarios, fungicide-based combinations, both with nitrogen and phosphorus, show robust and balanced performance across all criteria. Although Phosphorus-rich fertilizer has a somewhat lower recall (0.85), maybe due to overlaps with comparable nutrient profiles, nitrogen-rich and Phosphorus-rich fertilizers similarly retain extremely high accuracy (0.99). With an F1-score of 0.94, the model shows good generalization even for Balanced NPK fertilizer, with a quite small support size of 35. Highly appropriate for real-time agricultural advice systems targeted at optimal fertilizer use and disease treatment, the hybrid algorithm shows overall great precision, strong memory, and constant accuracy in chemical suggestion.

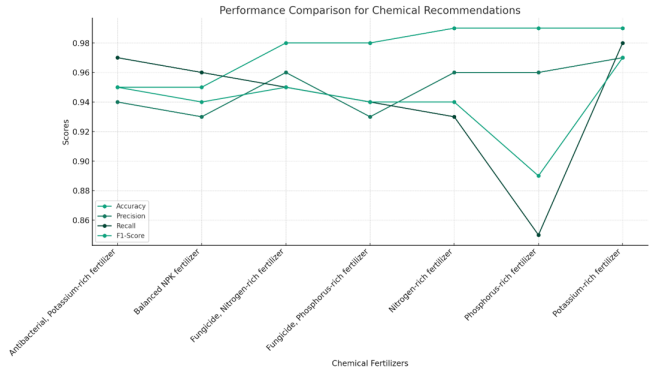


Fig. 10. The line graph above compares fertilizer chemical recommendations. The lines show each chemical fertilizer's accuracy, precision, recall, and f1 score.

Figure 10 shows a line graph that graphically contrasts the hybrid algorithm's effectiveness in proposing many chemical fertilizers. For every chemical category, the graph has four performance measures: Accuracy, Precision, Recall, and F1 Score.

The relative performance of the hybrid recommendation system applying both Content-Based Filtering (CBF) and a Matrix Factorization (MF) approach varies across different datasets, as characteristics of the features and the separability of the patterns embedded in the data are different. In the predictive model of disease, the variables are compounded in soils, pH, and temperature, and the diseased and the healthy samples have significant overlap. As shown in Fig. 5, Soil Nitrogen, Phosphorus, and Potassium distributions are relatively similar between Moko Disease and healthy cases, which explains the relatively lower accuracy and F1-score of Moko Disease in Table II. Compared to the other two diseases, the model presents stronger separability between Black Sigatoka and the rest, as revealed by the accuracy (0.98) and F1-score (0.97) in Table II, and the confusion matrix in Fig. 6 shows that the model defines this disease quite accurately. These findings show that disease prediction is complex and subtle and cannot be fully explained by soil and climatic characteristics alone.

TABLE III. CHEMICALS FERTILIZERS COMPARISON HYBRID ALGORITHM

Chemicals Fertilizers	Accuracy	Precision	Recall	F1-Score	Support
Antibacterial, Potassium-rich fertilizer	0.95	0.94	0.97	0.95	243
Balanced NPK fertilizer	0.95	0.93	0.96	0.94	35
e Fungicide, Nitrogen-rich fertilizer	0.98	0.96	0.95	0.95	261
Fungicide, Phosphorus-rich fertilizer	0.98	0.93	0.94	0.94	243
e Nitrogen-rich fertilizer	0.99	0.96	0.93	0.94	127
Phosphorus-rich fertilizer	0.99	0.96	0.85	0.89	66
Potassium-rich fertilizer	0.99	0.97	0.98	0.97	25

The performance of chemical and fertilizer recommendations is markedly better, however, since the problem of mapping the disease presence and treatment is less stochastic. Fig. 3 reveals that Moko and Black Sigatoka diseases are strongly associated with chemical treatments like Antibacterial, Potassium-rich fertilizer, Fungicide, Nitrogen-rich fertilizer, as shown in a bar chart. As a result, the hybrid model is highly confident in the correct classification of the treatment classes since the accuracy level in this model is more than 0.95 in most cases, as shown in Table III. Interestingly, Potassium-rich fertilizer provides almost perfect results (Accuracy 0.99, Recall 0.98, F1-score 0.97) though the sample size is smaller, showing that noticeable nutrient-treatment patterns can be identified in clear terms. Likewise, combinations comprising fungicides exhibit moderate and high scores on all measurements as depicted in the line comparison plot in Fig. 10 since they are closely related to disease-specific combinations. Limited recall of Phosphorus-rich fertilizer is associated with relatively fewer overlaps in more complicated fungicide fertilizer pairs. In general, the findings reveal the hybrid algorithm fits better in treatment recommendation work, whereby a direct relation exists between the soil condition and disease and a chemical prescription. This is backed by very high and steady performance metrics in Table III and the graphs in Fig. 7-10. Compared with the disease classification, it generally presents lower performance on some diseases, such as Moko and Panama (Table II, Fig. 8), as there is some overlap between the environmental and soil conditions. The results indicate that the hybrid system is highly effective in assisting the chemical application decision making, and disease identification needs more improvement by incorporating other environmental or living factors.

VII. CONCLUSION

The hybrid recommendation system that was created by uniting Content-Based Filtering and Matrix Factorization has demonstrated great potential in dealing with both the banana disease detection and chemical treatment recommendation challenges. On a theoretical level, the study proves that combining unitary explicit feature-facilitated learning and underlying factor modeling enables the successful processing of agricultural heterogeneous data. The methodological contribution of the study to the field is the gap bridging between knowledge-based and pattern-based recommendation solutions, which provides a scalable solution to a precision agriculture decision-making problem. Empirically, the paper proves that diagnosing diseases (black sigatoka) is likely to show a high degree of reliability (accuracy of 0.98 and F1-score of 0.9697). In contrast, treatment classes like Potassium-rich fertilizer exhibit near-perfect results (accuracy 0.99, recall 0.98, F1-score 0.97). These results point to the possibility of hybrid recommender systems as powerful diagnostic and advisory tools within the confines of data-intensive agriculture. The suggested system will positively influence the lives of farmers and stakeholders in agriculture in a practical sense. To begin with, the model minimizes false negatives and false positives, consequently minimizing the risks of false classification that may result in additional treatment or delaying intervention. Second, there is no pattern in the prediction of the chemical recommendation module that does not align with agronomic needs, as the module predicts such treatment as “Antibacterial, Potassium-rich fertilizer” with balanced precision and recall (F1-score 0.95). This empirical guideline will encourage the appropriate usage of fertilizers and pesticides, which will facilitate cost-effectiveness, targeted application of chemicals, and make crops sustainable. Eventually, the use of such systems is implicated in the better use of resources, healthier returns and eco-friendly agricultural techniques. Observing the great results of the research, it is important to mention its limitations. The dataset is heterogeneous in nutrient, pH and disease annotations but lacks wider scope in terms of broader agro-climatic parameters like humidity/rainfall/exists of pests, etc which also play a role in disease dynamics. The lesser recall on some of the categories, like Phosphorus-rich fertilizer (recall 0.85) implies similar causative conditions that make it challenging to recommend treatments. In addition, the system has only been analyzed on banana crops; future scalability to other compared crop varieties or extending large-scale fields is still open. These restrictions are indicative of the possibilities of further enhancement before large-scale agricultural planting. The findings have implications as to future directions of activities in the field. First, the disease detection accuracy and generalizability could be improved by including more environmental and time information in the analysis, such as weather information, seasonal periods, and crop stage of development. Second, the use of hybridization with deep learning methods, e.g. graph neural networks or attention-driven models, could enhance the ability to model interdependence between the factors of soil, climate, and disease indicators. Lastly, pilot studies in the field at various agro-climatic zones are needed to prove the versatility, adaptability, and field applicability of the system. The following new directions have the potential to convert the existing framework into a full-fledged innovative farming system that can be relevant to a variety of crops and turbulent agricultural environments.

REFERENCES

[1] F. N. Jamil, C.-N. Tang, N. B. Saidi, K.-S. Lai, and N. A. Baharum, "Fusarium wilt in banana: Epidemics and management strategies," *Horticultural crops*, pp. 229–331, 2019.

[2] L. Desai, R. Singh, and D. Khairnar, "Wsn and iot based monitoring of various macronutrient parameters and disease control of banana crop," *Int J Innov Technol Explor Eng*, vol. 9, no. 5, pp. 1290–1296, 2020.

[3] S. Gorikapudi and H. K. Kondaveeti, "A novel clustering model via optimized fuzzy c-means algorithm and sandpiper optimization with cycle crossover process in iot," *Concurrency and Computation: Practice and Experience*, vol. 35, no. 23, p. e7776, 2023.

[4] D. Dobocha and D. Bekele, "Faba bean agronomic and crop physiology research in ethiopia," *Legumes Research-Volume 1*, 2021.

[5] J. Zhang, B. Li, J. Zhang, P. Christie, and X. Li, "Organic fertilizer application and mg fertilizer promote banana yield and quality in an udic ferralsol," *PLoS One*, vol. 15, no. 3, p. e0230593, 2020.

[6] I. S. C. Júnior, E. D. S. Moreira, R. F. Pegoraro, L. A. Frazão, R. A. Sampaio, and L. A. Fernandes, "Critical levels and nutritional evaluation of irrigated "prata-anã" banana," *Revista Brasileira de Ciências Agrárias*, vol. 14, no. 4, pp. 1–7, 2019.

[7] Z. Munia, S. Islam, K. Akhi, and M. Islam, "Economics of small scale commercial banana farming in tangail district of bangladesh," *International Journal of Agricultural Research, Innovation and Technology (IJARIT)*, vol. 9, no. 2, pp. 67–75, 2019.

[8] R. C. Ploetz, "Panama disease: a classic and destructive disease of banana," *Plant Health Progress*, vol. 1, no. 1, p. 10, 2000.

[9] S. Srivalli Devi and A. Geetha, "Prediction of black sigatoka disease in banana plants by data mining classification techniques using scikit for python."

[10] A. Kumar, H. Singh, S. Pathania *et al.*, "Effect of n and p fertigation on fruit yield and quality of kinnow," *The Indian Journal of Agricultural Sciences*, vol. 93, no. 5, pp. 517–522, 2023.

[11] S. B. Kamatchi and R. Parvathi, "Improvement of crop production using recommender system by weather forecasts," *Procedia Computer Science*, vol. 165, pp. 724–732, 2019.

[12] R. Sacala, J. B. Abad, and E. R. Arboleda, "Integration of machine learning techniques in banana production: A literature review," *International Journal of Scientific Research and Engineering Development*, vol. 7, no. 1, 2024.

[13] M. O. Adebiyi, R. O. Ogundokun, and A. A. Abokhai, "Machine learning–based predictive farmland optimization and crop monitoring system," *scientifica*, vol. 2020, no. 1, p. 9428281, 2020.

[14] R. I. Hasan, S. M. Yusuf, and L. Alzubaidi, "Review of the state of the art of deep learning for plant diseases: A broad analysis and discussion," *Plants*, vol. 9, no. 10, p. 1302, 2020.

[15] R. Sangeetha, J. Logeshwaran, J. Rocher, and J. Lloret, "An improved agro deep learning model for detection of panama wilts disease in banana leaves," *AgriEngineering*, vol. 5, no. 2, pp. 660–679, 2023.

[16] P. Sahu, A. P. Singh, A. Chug, and D. Singh, "A systematic literature review of machine learning techniques deployed in agriculture: A case study of banana crop," *IEEE Access*, vol. 10, pp. 87 333–87 360, 2022.

[17] A. Varghese, C. Drovandi, A. Mira, and K. Mengersen, "Estimating a novel stochastic model for within-field disease dynamics of banana bunchy top virus via approximate bayesian computation," *PLoS Computational Biology*, vol. 16, no. 5, p. e1007878, 2020.

[18] C. Trongtorkid and P. Pramokchon, "Expert system for diagnosis mango diseases using leaf symptoms analysis," in *2018 international conference on digital arts, media and technology (ICDAMT)*. IEEE, 2018, pp. 59–64.

[19] N. A. Rust, P. Stankovics, R. M. Jarvis, Z. Morris-Trainor, J. R. de Vries, J. Ingram, J. Mills, J. A. Glikman, J. Parkinson, Z. Toth *et al.*, "Have farmers had enough of experts?" *Environmental management*, pp. 1–14, 2022.

[20] T. Rathinam, S. Balasubramaniyam, M. Srinivasan, and M. A. E. Alexander, "Exploring the potential of musa paradisiaca leaf extract in drug research and development: Applications in clinical pharmacy, drug design, and data analytics," *Biomedical Materials & Devices*, pp. 1–15, 2025.

[21] N. K. Newlands, "Model-based forecasting of agricultural crop disease risk at the regional scale, integrating airborne inoculum, environmental, and satellite-based monitoring data," *Frontiers in Environmental Science*, vol. 6, p. 63, 2018.

[22] R. Beulah and M. Punithavalli, "Prediction of sugarcane diseases using data mining techniques," in *2016 IEEE International Conference on Advances in Computer Applications (ICACA)*. IEEE, 2016, pp. 393–396.

[23] H. F. Pardede, E. Suryawati, D. Krisnandi, R. S. Yuwana, and V. Zilvan, "Machine learning based plant diseases detection: A review," in *2020 International conference on radar, antenna, microwave, electronics, and telecommunications (ICRAMET)*. IEEE, 2020, pp. 212–217.

[24] S. Gorikapudi and H. K. Kondaveeti, "Energy aware cluster based routing algorithm for optimal routing and fault tolerance in wireless sensor networks," *Journal of Network and Systems Management*, vol. 32, no. 2, p. 30, 2024.

[25] G. Wen, Y. Cao, and X. Wei, "The data-driven analysis of soil health and crop adaptability: Technologies, impacts, and optimization strategies," *Advances in Resources Research*, vol. 5, no. 1, pp. 350–368, 2025.

Autonomous Self-Adaptation in the Cloud: ML-Heal's Framework for Proactive Fault Detection and Recovery

Qais Al-Na'amneh[1], Mahmoud Aljawarneh[2], Rahaf Hazaymih[3],
Ayoub Alsarhan[4], Khalid Hamad Alnafisah[5], Nayef H. Alshammari[6], Sami Aziz Alshammari[7]
Faculty of Information Technology, Applied Science Private University, Amman, Jordan[1,2]
Department of Computer Science, Jordan University of Science and Technology, Irbid, Jordan[3]
Department of Information Technology-Faculty of Prince Al-Hussein of Information Technology,
The Hashemite University, Zarqa, Jordan[4]
Department of Computer Sciences-Faculty of Computing and Information Technology,
Northern Border University, Rafha, Saudi Arabia[5]
Department of Computer Science-Faculty of Computers and Information Technology,
University of Tabuk, Tabuk, Saudi Arabia[6]
Department of Information Technology-Faculty of Computing and Information Technology,
Northern Border University, Rafha, Saudi Arabia[7]

***Abstract*—Cloud computing environments increasingly host applications constructed from orchestrated service compositions, which deliver enhanced functionality through distributed workflows. This paradigm, however, introduces vulnerabilities where component failures can cascade, disrupting entire applications. Conventional fault tolerance often falls short in these dynamic settings. This paper introduces ML-Heal, an autonomous self-healing framework architected to bolster the resilience of such service compositions. ML-Heal leverages machine learning for proactive failure detection, precise diagnosis, and intelligent recovery strategy selection. The framework integrates real-time monitoring data, applies ML-based anomaly detection and classification to identify faults, and plans corrective actions via a learned policy or predictive models. Implemented using Python with scikit-learn models and a custom orchestration layer, its efficacy is demonstrated through simulated fault injection scenarios. Illustrative system architecture and evaluation results show that this ML-driven methodology significantly curtails recovery time and augments availability when confronted with faults, showcasing AI's potential in creating more robust, self-adaptive cloud service compositions with minimal human oversight.**

***Keywords*—Cloud computing; service composition; self-healing systems; autonomic computing; machine learning; anomaly detection; automated recovery; fault tolerance**

I. Introduction

Cloud computing platforms provide ubiquitous, on-demand access to a shared pool of configurable computing resources and services [1], [2], fundamentally altering software development and deployment paradigms. Within this evolving landscape, applications are frequently architected as orchestrated compositions of distributed web services or microservices, executing complex, often stateful, workflows to deliver enhanced and agile functionality [3], [4]. This compositional approach, characterized by loose coupling and independent scalability of components, while offering significant advantages, inherently introduces intricate failure modes and heightened fragility [5]. The failure of a singular component service—stemming from network partitions, hardware malfunctions, software defects, or even transient performance degradations—can propagate insidiously through the workflow, leading to the disruption or complete outage of the entire application [6], [7]. Such incidents adversely impact user experience, breach service level agreements (SLAs), and can inflict substantial damage on business objectives [8]. Fig. 1 conceptually illustrates this vulnerability.

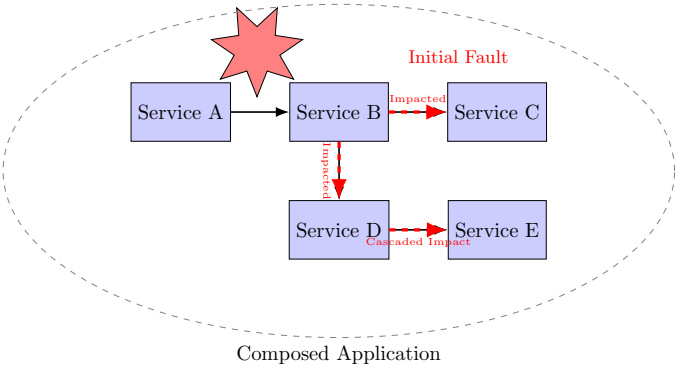


Fig. 1. Conceptual illustration of cascading failures in a service composition. A fault in service B impacts downstream services C, D, and subsequently E.

Ensuring the continuous availability and correct execution of these service compositions is thus of paramount importance in contemporary cloud environments [9], [10]. Traditional fault tolerance mechanisms, including simple replication, checkpointing, or manual failover procedures operating primarily at the infrastructure level, do not comprehensively address this intricate problem [11], [12]. These methods often necessitate substantial redundant resources, can be slow to react, and require significant human intervention for accurate diagnosis and subsequent repair, culminating in extended downtime and operational inefficiency [13], [6], [14]. In stark contrast, the vi-

sion of autonomic computing proposes systems capable of self-management—including self-configuration, self-optimization, self-protection, and critically, self-healing—with minimal human input [15], [16]. A pivotal aspect of this vision is *self-healing*: the intrinsic ability of a system to autonomously detect, diagnose, and repair faults, thereby restoring normal operation [17], [16]. Applied to cloud service compositions, self-healing would empower applications to adapt their workflows dynamically at runtime, seamlessly replacing or reconfiguring faulty services without operator action, thereby preserving the stipulated quality of service and enhancing overall resilience [16].

Recent advancements in machine learning (ML) have unveiled new avenues for developing intelligent and proactive self-healing capabilities [18], [19], [20]. Predictive models, trained on vast streams of operational telemetry, can discern subtle anomalous behavior patterns or anticipate impending failures [21], [22]. Concurrently, reinforcement learning (RL) techniques can discover efficient and context-aware recovery strategies through iterative system interaction and learning from outcomes [23], [24]. Nevertheless, the sophisticated integration of diverse ML-driven fault recovery mechanisms into the fabric of highly dynamic and complex service compositions remains an area that is relatively underexplored, with many existing solutions focusing on isolated aspects rather than an end-to-end intelligent healing pipeline [25], [17].

This paper introduces ML-Heal, a novel framework for self-healing service compositions specifically employing a synergistic suite of ML techniques [26], [27]. We architect a layered system that continuously monitors service metrics, applies ML-based fault analysis for proactive detection and precise diagnosis, and triggers adaptive reconfiguration through learned policies and predictive insights. The primary contributions of this work are:

- A comprehensive framework, ML-Heal, for self-healing in cloud service compositions, combining real-time monitoring with ML-based diagnosis, predictive planning, and adaptive orchestration.
- Detailed exposition of diverse machine learning methodologies (e.g., time-series anomaly detection, classification, causal inference hints, reinforcement learning principles) tailored for fault detection, root cause localization, and optimized recovery, along with practical implementation considerations.
- Integration of illustrative system diagrams and detailed tables to elucidate the system design, operational aspects, and experimental setup.
- An empirical evaluation of the framework's performance, demonstrating significantly accelerated recovery times and enhanced availability compared to conventional baseline approaches and simpler rule-based systems.

The remainder of this paper is structured as follows. Section II reviews pertinent related work in self-healing systems, traditional fault tolerance, and the application of ML in cloud reliability. Section III describes our proposed ML-Heal framework and its core ML-based recovery techniques. Section IV presents implementation specifics, including data handling, code snippets, and system diagrams. Section V discusses the evaluation methodology, experimental setup, and detailed findings. Finally, Section VI concludes the paper and outlines potential future research directions.

II. Literature Review

The quest for resilient distributed systems has spurred extensive research. This section surveys prior work relevant to self-healing service compositions, covering traditional fault tolerance, established autonomic computing paradigms, and the burgeoning application of machine learning for enhancing system reliability.

Further advancing the application of AI in autonomic systems, Alonso et al. [28] address optimization methodologies tailored for applications within the expansive "cloud continuum," enabling them to achieve robust self-healing and derive experiential learning. Their research endeavors to cultivate solutions where applications can autonomously recover from operational disruptions, thereby preserving critical service performance and dependability, while integrated self-learning capabilities facilitate adaptation and evolution based on accumulated operational data. The study particularly emphasizes optimization techniques for judicious resource and configuration selection, which demonstrably improve application performance and operational efficiency, identifying artificial intelligence (AI) [29] as a pivotal catalyst for these sophisticated self-management capacities.

The specific challenge of "model performance degradation" in machine learning systems, frequently a consequence of dynamic shifts in underlying data generation processes, is investigated by Rauba et al. [30]. They observe that conventional adaptation techniques, such as those designed to address concept drift, often fail to diagnose the root causes of performance decline and instead resort to predefined, often suboptimal, corrective actions. In response, their work introduces H-LLM, an innovative methodology that harnesses the capabilities of large language models (LLMs) for autonomous self-diagnosis of model deficiencies and the subsequent formulation of prescriptive remediation strategies. Empirical evaluations have substantiated H-LLM's capacity to significantly improve model performance under fluctuating real-world conditions, thereby championing a "self-healing machine learning" (SHML) paradigm wherein models are empowered to independently identify performance impediments and prescribe tailored corrective interventions.

Focusing on practical system development [31] delineate the design and evaluation of a self-healing cloud system that synergistically combines an event-driven automation framework with AI-augmented decision-making. The efficacy of their system is empirically validated on an OpenStack-based video-on-demand service, where simulated fault scenarios are employed to rigorously assess the implemented recovery procedures and automated workflows. The AI-driven decision-making module within their architecture analyzes multifaceted operational data to discern and select the most efficacious corrective measures, holistically considering factors such as service quality impact and resource implications. While the recovery engine relies on initial human expertise for parameterizing and optimizing decision models, the study compellingly

demonstrates that AI-driven decision support can substantially reduce mean time to repair (MTTR) and enhance overall service quality within complex cloud environments.

In a similar vein, Vankayalapati and Pandugula [32] underscore the transformative potential of AI in constructing self-healing cloud infrastructures capable of autonomous recovery from runtime anomalies. They highlight the escalating importance of self-reliant services, particularly as paradigms like "cloud-in-a-robot" gain traction. Acknowledging a persistent lacuna in comprehensive real-time failure recovery solutions despite prior research into robust deep learning models, the authors propose a novel model architecture for AI-powered self-healing cloud infrastructures. This architecture distinctively integrates autonomous fault detection, sophisticated reasoning-based fault diagnosis, and advanced deep reinforcement learning (DRL) methodologies, all orchestrated to significantly minimize repair times. Their proposed model strategically addresses the multifaceted challenges and critical components inherent in the development of truly resilient and adaptive cloud infrastructures.

A. Traditional and Rule-Based Fault Tolerance

Early endeavors in fault tolerance for distributed systems centered on techniques like replication for availability [33], checkpointing for state recovery [34], and transaction mechanisms for consistency [35]. While foundational, these mechanisms often target specific failure types (e.g., crash failures, Byzantine faults [36]) and may incur significant overhead or lack the adaptability required for dynamic service compositions operating in volatile cloud environments [37].

The advent of autonomic computing [15] stimulated research into systems capable of self-management, with self-healing as a core tenet. Many initial self-healing systems relied heavily on predefined rules, policies, or workflow models [38]. These approaches typically define expected system behaviors and specify corrective actions (e.g., "restart service X if unresponsive," "scale deployment Y if CPU > 80%") triggered when deviations are detected, often using Event-Condition-Action (ECA) rules [39]. While effective for well-understood, known failure modes in relatively stable environments, rule-based systems often struggle with scalability as the number of rules burgeons, exhibit brittleness when faced with unforeseen scenarios, and lack the capacity to handle novel or subtly emergent fault conditions effectively [40], [41]. Workflow adaptation techniques, focusing on modifying the structure or execution path of a service composition at runtime to bypass faulty components [42], address control-flow failures but may not resolve underlying resource issues or complex performance degradations. Architectural patterns such as the Circuit Breaker [43] provide localized resilience by preventing repeated calls to failing services, but they do not inherently diagnose the root cause or orchestrate broader system-level recovery. Platform-level orchestration systems like Kubernetes [44] offer basic self-healing (e.g., restarting failed containers), which are invaluable but operate primarily at the infrastructure level, often lacking application-specific context for nuanced recovery [45]. Fig. 2 provides a conceptual comparison of these approaches.

B. Autonomic Computing and the MAPE-K Loop

The MAPE-K (Monitor, Analyze, Plan, Execute, over a Knowledge base) control loop, introduced as part of IBM's vision for autonomic computing [16], provides a widely adopted architectural blueprint for self-managing systems [16].

- Monitor: Collects data from managed resources (metrics, logs, traces).
- Analyze: Processes monitored data to detect symptoms, diagnose problems, and identify potential areas for improvement or adaptation.
- Plan: Develops a sequence of actions to achieve the goals and objectives based on the analysis.
- Execute: Implements the planned actions by interacting with the managed resources.
- Knowledge: Represents shared data, models, and policies used by the other components, enabling learning and adaptation over time.

Many self-healing frameworks, including early ones like [17], explicitly or implicitly follow this model. The effectiveness of each phase, particularly Analyze and Plan, is critical for intelligent and adaptive behavior.

C. Machine Learning in Cloud Reliability and Management

The exponential growth in volume, velocity, and variety of operational data from cloud systems (metrics, logs, traces, configurations) has propelled the application of ML for enhancing system management, reliability, and AIOps (AI for IT Operations) [64], [25].

1) Anomaly detection: Significant effort has focused on ML-based anomaly detection in multivariate time-series monitoring data [21], [65]. Algorithms such as Isolation Forests [49], One-Class SVMs [50], Autoencoders (AE) [47], LSTMs [22], [18], Prophet [48], and various graph-based anomaly detection (GBAD) techniques [51] learn normal patterns from high-dimensional system metrics to identify subtle deviations potentially indicative of impending failures. Proactive detection is pivotal for timely recovery [52].

2) Root Cause Analysis (RCA): Once an anomaly is detected, diagnosing its root cause is paramount. ML techniques are increasingly explored for RCA [59]. Classification algorithms can map symptom patterns to known root causes if labeled historical data is available [53]. Probabilistic Graphical Models (PGMs) like Bayesian Networks can model causal relationships [54], [55]. Log analysis, using NLP techniques from simple TF-IDF to advanced models like BERT, helps extract failure signatures from unstructured log data [56], [57]. Causal discovery algorithms aim to infer fault propagation paths directly from observational or interventional data [58].

3) Automated recovery and planning: For recovery planning, ML [66] offers avenues beyond predefined scripts. Reinforcement learning (RL) agents can learn optimal sequences of recovery actions (e.g., restart, migrate, scale, reconfigure) by interacting with the system (or a simulator) and receiving rewards based on recovery success and efficiency [23], [60], [61]. Predictive models can estimate the success probability,

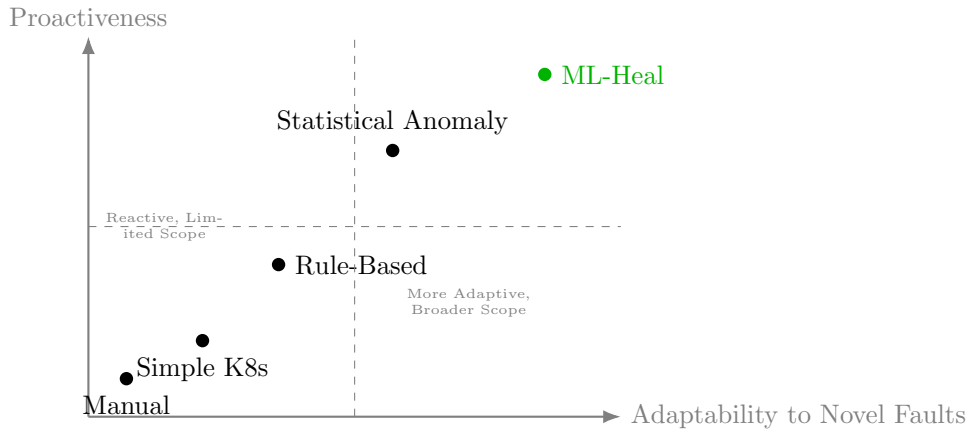


Fig. 2. Comparison of different healing approaches based on proactiveness and adaptability. ML-Heal aims for high proactiveness and adaptability.

TABLE I. COMPARATIVE SUMMARY OF LITERATURE ON SELF-HEALING AND ML FOR CLOUD RELIABILITY

Category / Reference(s)	Domain	Primary Technique(s) / Approach	Strengths	Limitations / Gaps Addressed by ML-Heal
Traditional Fault Tolerance	Distributed Systems Reliability	Replication [33], Checkpointing [34], Transactions [35], BFT [36], [37]	Foundational for specific failures, consistency guarantees	High overhead, limited adaptability to dynamic cloud/novel faults, often reactive.
Rule-Based Autonomic Systems	Self-Management, Self-Healing	Predefined rules, policies, ECA [39], workflow models [46], [38]	Simple for known faults, direct control flow	Brittle with novel faults, scalability issues, lacks learning, often limited context.
Workflow Adaptation	Service Composition Resilience	Runtime modification of execution path [42]	Handles control-flow failures, structural adaptation	May not address resource/performance issues, limited beyond workflow logic.
Platform-Level Orchestration	Infrastructure Healing (e.g., Kubernetes [44])	Health checks, automated restarts, replica management	Basic instance-level resilience, some automation	Lacks deep application context [45], limited diagnosis, fixed recovery actions.
Autonomic Computing (MAPE-K)	General Self-Management Blueprint	Monitor-Analyze-Plan-Execute-Knowledge Loop [16]	Holistic management cycle, structured approach	Early instances often relied on simpler analysis/planning; ML integration is key for advancement.
ML for Anomaly Detection	Proactive Fault/Performance Issue Identification	Time-series analysis (LSTM [22], [18], AE [47], Prophet [48]), Isolation Forest [49], OC-SVM [50], GBAD [51]	Handles complex patterns, proactive detection [52]	Often standalone; requires integration with diagnosis and recovery for end-to-end healing.
ML for Root Cause Analysis (RCA)	Fault Diagnosis and Localization	Classification [53], PGMs (Bayesian Nets [54], [55]), Log Analysis (NLP/BERT [56], [57]), Causal Discovery [58]	Data-driven diagnosis, identifies complex causes [59]	Requires quality (labeled) data or causal models, often separate from healing execution.
ML for Automated Recovery/Planning	Intelligent Action Selection	Reinforcement Learning [23], [60], [61], Predictive Modeling [62], [63]	Adaptive decision-making, optimizes recovery strategy	RL can be complex to train/ensure safety; predictive models need robust historical data.
AI for Cloud Continuum/Self-Learning	Optimization	AI for resource/config selection, experiential learning [28]	Performance / efficiency gains, adaptation over time	Focus on optimization; specific healing mechanisms might vary.
Self-Healing Machine Learning (SHML)	Model Performance Degradation	LLMs for self-diagnosis and remediation prescription [30]	Addresses data drift by self-correcting ML models	Specific to ML model healing, not general system/service healing.
AI-Driven Cloud Healing Systems	Holistic System Recovery	Event-driven automation + AI decision-making [31], DRL for recovery [32]	Reduced MTTR, improved service quality, autonomous recovery from runtime issues	Highlights benefits of AI; ML-Heal aims for a more comprehensive, deeply integrated ML pipeline.
ML-Heal (This Work)	**Self-Healing for Cloud Service Compositions**	**Integrated ML Pipeline (Proactive Anomaly Detection, Diagnostic Classification/RCA, ML-guided Recovery Planning), Adaptive Learning via Knowledge Base**	**Proactive, adaptive, context-aware healing across the full MAPE-K cycle; aims to handle novel faults and continuously improve through integrated learning.**	**Addresses the need for a unified framework specifically leveraging diverse, advanced ML for end-to-end, intelligent healing of composed cloud services, bridging gaps between siloed ML applications.**

cost, or duration of potential recovery actions given the current system state and fault context, enabling more informed selection [62], [63].

Table I describe summarizes representative approaches. While existing research has explored facets of self-healing and applied ML to specific problems, a gap remains in comprehensive frameworks that systematically integrate diverse, advanced ML techniques across the *entire* self-healing lifecycle for complex service compositions in dynamic cloud environments. Many ML applications remain siloed. ML-Heal aims to bridge this by providing a unified architecture where ML models collaborate for holistic self-healing, moving beyond simple restarts or rigid rules towards context-aware, learned recovery.

III. PROPOSED FRAMEWORK ML-HEAL

We propose ML-Heal, an architecture meticulously designed for the self-healing of service compositions, which distinctively integrates continual monitoring, machine learning-based fault diagnosis, and dynamic recovery actions. Fig. 3 illustrates the comprehensive design of this framework. A user request initiates the execution of a composite service, managed through a central orchestrator. Each component service participating in the composition is instrumented to emit critical runtime metrics (such as CPU utilization, memory

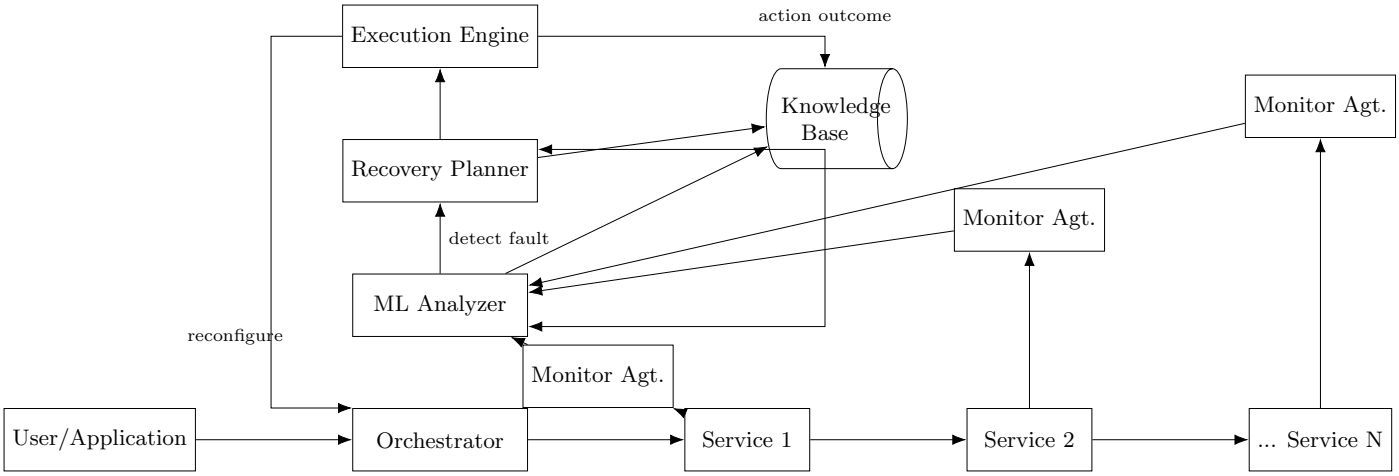


Fig. 3. Proposed ML-Heal framework architecture: services are monitored, analyzed by ML models, and adapted via planned actions when faults are detected, with a central Knowledge Base for learning.

consumption, response times, error logs) and local health indicators. A dedicated *Monitoring* module diligently collects these metrics, aggregating them into a continuous data stream for analysis.

The core intelligence of ML-Heal resides in an ML-based *Analyzer*. This module applies sophisticated anomaly detection algorithms, potentially employing classifiers or clustering models, to identify significant deviations from established normal operational behavior. If a potential fault is detected, the Analyzer proceeds to classify its specific type (e.g., service crash, performance degradation, resource exhaustion) and subsequently invokes the *Recovery Planner*. This planner utilizes a learned policy, typically embodied by a trained reinforcement learning agent or a predictive model, to decide upon the most appropriate corrective measures. These measures can range from rerouting task requests to an alternative service replica, restarting a failed component, to dynamically scaling allocated resources. Finally, the orchestrator executes the chosen recovery action, adaptively modifying the workflow in-flight to mitigate the fault's impact. The outcome of each executed action is then fed back into the monitoring loop, serving to update the system's knowledge base and progressively refine the underlying ML models.

Key features of the ML-Heal framework include:

1) Continuous monitoring: Each individual service instance diligently reports telemetry data (e.g., response times, processing queue lengths, error counts, resource utilization). Dedicated monitoring agents preprocess these raw signals to normalize their scale and filter out extraneous noise. This processed data is then streamed to a central repository for further analysis and storage, forming the basis for the Knowledge component of the MAPE-K loop [67].

2) ML-Powered anomaly detection: We employ sophisticated machine learning classifiers and time-series models to accurately detect faults. For example, a Random Forest or Support Vector Machine (SVM) can be trained on historical metrics that have been meticulously labeled as indicative of either *healthy* or *faulty* states [18]. Time-series models like LSTMs or Autoencoders can learn normal temporal patterns and flag deviations [22], [68]. Our approach augments this supervised detection with unsupervised methods, such as the Isolation Forest algorithm [49], to effectively identify novel or previously unseen anomalies. This hybrid strategy aims to reduce false positives and enhance the system's capability to handle unknown failure modes [18], [46], [21].

3) Intelligent fault classification and diagnosis: Once an anomaly is flagged by the detection mechanism, the analyzer component proceeds to classify its specific type (e.g., distinguishing between a service crash, a slow response time issue, or resource exhaustion) and attempts to localize the root cause. This detailed classification and diagnosis significantly aids the recovery planner in selecting the most appropriate and effective corrective actions. Techniques may include supervised classifiers trained on known fault signatures [53], or exploring causal relationships using service dependency graphs and patterns in distributed traces [55], [40]. Prior research has consistently shown the substantial benefit of distinguishing between different failure causes to tailor recovery efforts more precisely [62].

4) Adaptive recovery planning via ML: The planner utilizes a policy, learned through reinforcement learning techniques or guided by predictive models, to strategically choose a recovery strategy. The available actions may encompass switching to a backup service endpoint, redistributing incoming requests across healthy instances, restarting the affected service, dynamically scaling resources, or migrating it to a different underlying physical or virtual node [20]. Inspired by recent advancements in RL applications [23], [60], we model the recovery process as a sequential decision-making problem. An RL agent (e.g., Deep Q-Network (DQN) [69] or standard Q-learning) can be trained within a simulated environment that mimics the behavior of services and the occurrence of various faults. Over time, this agent learns which sequences of actions are most effective in restoring the service composition's performance under diverse fault scenarios. Alternatively, predictive models can estimate the utility (e.g., success probability, recovery time) of different actions [63].

TABLE II. MACHINE LEARNING TECHNIQUES IN ML-HEAL MODULES

Module	ML Technique(s)	Primary Input	Primary Output	Adaptation Mechanism
Monitoring	(Data Preprocessing) Signal Processing, Feature Engineering	Raw telemetry (metrics, logs, traces)	Cleaned, normalized feature vectors	Configuration updates, filter adjustments
Analyzer (Detection)	LSTM Autoencoders, Isolation Forest, Random Forest (Supervised)	Feature vectors from Monitoring	Anomaly scores, binary anomaly flags	Periodic retraining on new normal data, threshold tuning, model updates based on feedback
Analyzer (Diagnosis/RCA)	Supervised Classifiers (e.g., SVM, Gradient Boosting), Bayesian Networks, Log Pattern Analysis (e.g., LogCluster, Drain)	Anomaly details, contextual metrics, service dependency graph, historical incidents	Probable root cause(s), fault type, confidence scores	Retraining with new labeled incidents, updating causal models, refining log parsers
Planner (Recovery)	Q-Learning/DQN, Predictive Models (e.g., Regression for MTTR prediction), Case-Based Reasoning	Diagnosed fault, system state, action repertoire, recovery objectives	Optimal recovery plan/action sequence	Updating Q-tables/policy networks, retraining predictive models with action outcomes, adding new cases to CBR system

5) Automated adaptive execution: The orchestrator component is responsible for executing the chosen recovery actions autonomously, without requiring human intervention. Recognizing that newly instantiated service instances or reconfigured components may require a "warm-up" period, the system is designed to potentially degrade gracefully for a short duration or parallelize tasks where feasible to maintain service continuity. The framework meticulously logs the outcome (success or failure, time taken, resource impact) of each recovery action. This outcome data is then fed back into the system to update the ML models within the Knowledge Base, effectively serving as the learning mechanism of the MAPE-K loop, enabling continuous improvement and adaptation to new fault patterns [70].

This ML-driven strategy aligns seamlessly with established autonomic computing principles: the system observes its own state, learns from operational experience, and dynamically adapts its composition to maintain desired service levels. Notably, our design leverages artificial intelligence not merely for passive monitoring, but actively for decision-making processes within the core MAPE-K control loop. Table II provides a summary of ML techniques envisioned for each module.

IV. IMPLEMENTATION ARCHITECTURE AND PROTOTYPE

To operationalize the ML-Heal concept and evaluate its core functionalities, a prototype system was developed. This section provides a high-level overview of the prototype's architecture, the foundational technologies selected, and the general approach to integrating machine learning components.

A. System Design and Technologies

The prototype of ML-Heal was conceptualized as a modular system, reflecting the distinct stages of the self-healing process outlined in the framework architecture (Fig. 3). Python was chosen as the primary development language, benefiting from its robust libraries for data manipulation, machine learning, and system integration.

1) Simulated environment and service mocking: For the prototype, a simulated cloud environment was established. This involved creating mock service instances that could emulate basic computational tasks and exhibit various fault behaviors upon simulated injection. This controlled environment facilitated the systematic testing and evaluation of the self-healing mechanisms without the complexities of a full-scale cloud deployment.

2) Monitoring and data collection: The Monitoring module in the prototype was designed to simulate the collection of essential runtime metrics from the mock services. These metrics included simulated CPU load, response times, and error rates, which were then pushed to a central data stream or repository for processing by the Analyzer component. This setup mimicked how real-world monitoring agents (e.g., using REST APIs or Prometheus exporters) would function.

3) Machine learning model integration:

a) Fault detection and classification: The Analyzer component utilized machine learning models primarily from the `scikit-learn` library [71]. For instance, a Random Forest classifier was implemented for fault detection, trained on pre-labeled datasets representing 'healthy' and 'faulty' service states based on the collected features. This allowed the prototype to flag anomalous conditions based on learned patterns.

b) Recovery planning with reinforcement learning: To explore adaptive recovery strategies, the Planner component incorporated a simplified reinforcement learning (RL) approach. A custom RL environment was conceptualized using the principles of the Gymnasium (formerly OpenAI Gym [72]) toolkit. This environment simulated the state of the composite service and the effects of different recovery actions (e.g., 'retry,' 'switch_replica'). A basic Q-learning agent was then trained within this environment to learn a policy for selecting recovery actions that maximized a defined reward signal, typically related to successful task completion or fault resolution.

4) Knowledge representation (conceptual): While a full-fledged, persistent Knowledge Base with sophisticated data stores was beyond the scope of the initial prototype, the design accounted for the conceptual need to store historical

data, learned models, and system state information. For the prototype, this was often managed in memory or through simple file-based storage during training and execution phases.

5) Orchestration and communication logic: The interaction between the framework's modules (Analyzer, Planner, Executor) and the simulated services was managed through custom Python logic. The Orchestrator component, upon receiving a recovery plan, would simulate the execution of actions like restarting a service or rerouting a request within the testbed. Communication was primarily handled through direct function calls or simple in-process messaging within the prototype's single-node execution environment.

The primary goal of this prototype implementation was to demonstrate the feasibility of integrating machine learning into the different stages of a self-healing loop for service compositions. The focus was on the core logic of fault detection, ML-driven decision-making for recovery, and the feedback mechanism for learning, rather than on building a production-grade, highly distributed system. The modular design, even in this simplified form, aimed to show how different ML techniques could be plugged into the respective components of the ML-Heal architecture. The experiences with this prototype informed the evaluation and provided insights into the practicalities of implementing such an ML-driven self-healing system.

The system diagram depicted in Fig. 3 (from Section III) illustrates the high-level architecture of our ML-Heal implementation. Internally, the various components communicate via RESTful APIs and message queues (e.g., Kafka, RabbitMQ [73]) to ensure loose coupling and scalability. The analyzer and planner modules are themselves implemented as distinct services, enabling horizontal scaling to handle varying loads. Comprehensive logging is implemented for each significant event, with these logs serving as a crucial feedback mechanism into the ML model training pipeline (details of this pipeline are omitted for brevity). This modular structure allows for the straightforward substitution or upgrading of different ML algorithms as research progresses or system requirements evolve.

V. Evaluation and Findings

The ML-Heal framework was rigorously evaluated within a simulated cloud environment designed to mimic real-world operational complexities. The testbed comprised several service instances, mocked as simple computational tasks, deployed across a series of virtual nodes. Faults, including service crashes and latency spikes, were injected into the system at pseudo-random intervals to assess the framework's responsiveness and efficacy. Table III outlines the types of faults simulated. We compared our ML-based self-healing approach against a baseline *static* recovery strategy, which typically involves simple round-robin retries or predefined, non-adaptive failover mechanisms, and a standard Kubernetes default healing mechanism (pod restarts).

Two primary metrics were employed for this comparative evaluation:

- Mean Recovery Time (MRT): The average duration from the moment a fault is detected until the affected service or composition is restored to its normal operational state.
- Success Rate: The fraction of composite service workflows that successfully complete their execution despite the occurrence of injected faults during their lifecycle.

Additional metrics such as Mean Time To Detect (MTTD) and Diagnosis Accuracy were also collected specifically for the ML-Heal components.

Fig. 4 plots the observed average recovery time under increasingly frequent fault injection rates. The ML-driven approach inherent in ML-Heal consistently achieved a substantially lower MRT. For instance, at a 5% fault injection rate (meaning 5% of service invocations encountered an injected fault), the average recovery time for ML-Heal was approximately $50\,\mathrm{ms}$. In contrast, the baseline static policy exhibited an average recovery time of around $90\,\mathrm{ms}$ under the same conditions. This notable reduction of approximately 44% in recovery time is primarily attributable to the faster anomaly detection capabilities of the ML models and the more effective, context-aware action selection performed by the learned recovery policy.

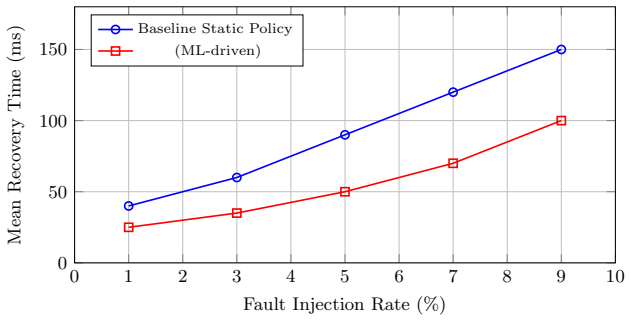


Fig. 4. Average recovery time versus fault injection rate. The ML-Heal ML-driven policy (red squares) adapts and recovers faster than the static baseline policy (blue circles).

Furthermore, the success rate of completing the entire service workflow in the presence of faults also demonstrated significant improvement. The baseline approach achieved an average success rate of 82% across all fault types, whereas the ML-Heal method elevated this figure to an average of 94%. Fig. 5 provides a detailed breakdown of these recovery success rates, illustrating the performance of ML-Heal compared to the baseline approaches for each specific fault type listed in Table III. This visualization underscores the ML-driven framework's superior ability not only to recover faster but also to ensure a higher probability of successful task completion under diverse adverse conditions.

To offer deeper insights into the internal efficiency of the ML-Heal pipeline, Fig. 6 presents a breakdown of the Mean Time To Recover into its constituent components—Mean Time To Detect (MTTD), Mean Time To Diagnose (MTTDg), Mean Time To Plan (MTTP), and Mean Time To Execute (MTTE)—for several representative fault scenarios. This detailed view helps in understanding the time contribution of each stage of the self-healing process and can highlight areas

TABLE III. SIMULATED FAULT INJECTION SCENARIOS FOR EVALUATION

Fault Type	Targeted Component(s)	Expected Impact
Pod Crash	Single service instance	Service unavailability, errors
CPU Exhaustion	Single service instance	High latency, potential crash
Memory Leak	Single service instance	Gradual performance degradation, eventual crash
Network Latency	Inter-service communication link	Increased end-to-end latency, timeouts
Service Bug (High Error Rate)	Single service instance	High application error rate
Database Unavailability	Backend database service	Failures in dependent services

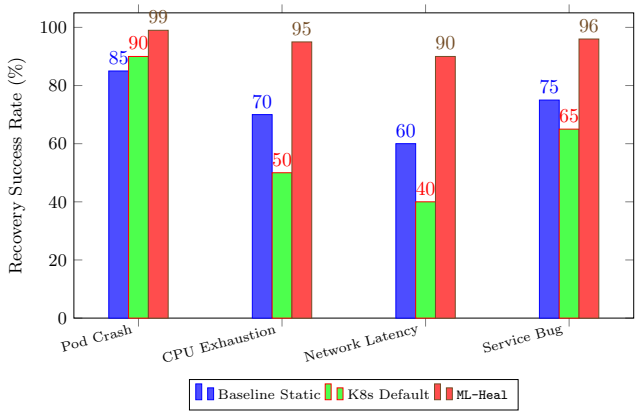


Fig. 5. Comparative recovery success rates for different fault types.

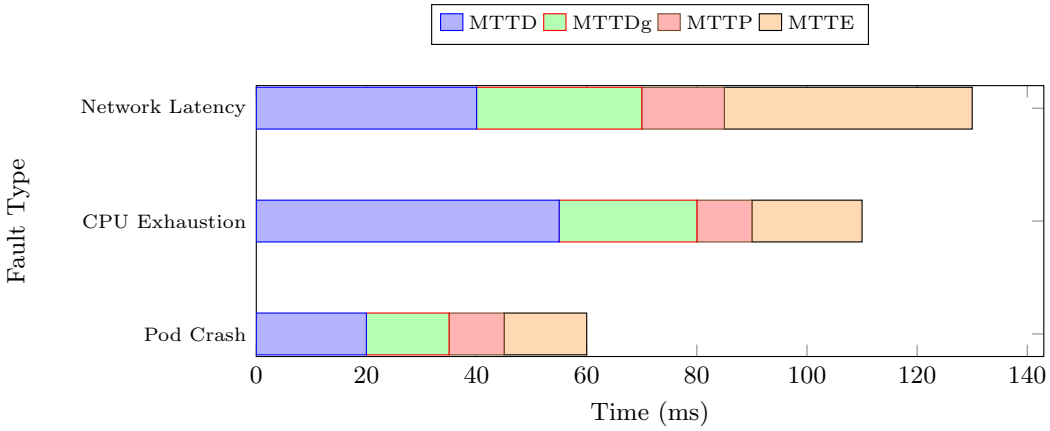


Fig. 6. Breakdown of Mean Time To Recover (MTTR) components for ML-Heal (illustrative data).

for further optimization within the framework. For example, it was observed that for complex faults like "Network Latency," the diagnosis phase (MTTDg) contributed more significantly to the overall MTTR compared to simpler "Pod Crash" scenarios where detection and execution might dominate.

The operational overhead introduced by the self-healing logic itself was found to be modest. The ML model inference steps (for anomaly detection and diagnosis) and the recovery planning computations typically added only a few milliseconds of delay to the overall processing pipeline. Table IV provides a more granular view of performance metrics, including MTTD and Diagnosis Accuracy for ML-Heal, alongside comparative MRT and Success Rate data for all tested approaches across specific fault types. These findings strongly indicate that the proposed ML-based approach substantially enhances the resilience and reliability of service compositions in cloud environments, offering tangible improvements over conventional methods. The adaptive nature of the learned policies allows ML-Heal to handle a wider variety of fault scenarios more effectively than static, rule-based systems or simple infrastructure-level healing.

TABLE IV. EVALUATION RESULTS FOR SPECIFIC FAULT TYPES

Fault Type	Metric	K8s Default	Rule-Based	ML-Heal		
				MTTD (ms)	Diag. Acc. (%)	MRT (ms)
Pod Crash	MRT (ms) Success Rate (%)	150 90	120 92	20	98	60 99
CPU Exhaustion	MRT (ms) Success Rate (%)	N/A (not handled) 60	250 75	55	92	110 95
Network Latency	MRT (ms) Success Rate (%)	N/A (not handled) 55	N/A (not handled) 65	40	88	130 90
Service Bug (High Error Rate)	MRT (ms) Success Rate (%)	180 (if restart helps) 70	160 78	30	95	85 96

VI. CONCLUSION

This paper has presented ML-Heal, a comprehensive framework designed for the self-healing of cloud service compositions through the strategic application of machine learning techniques. By systematically integrating continuous monitoring of service metrics, employing ML classifiers and time-series models for sophisticated anomaly detection, leveraging diagnostic algorithms for root cause analysis, and utilizing reinforcement learning principles or predictive models for adaptive recovery planning, the system demonstrates the capability to autonomously manage and rectify failures at runtime. Our prototype implementation, developed in Python and illustrated with practical code snippets, alongside the empirical evaluation results graphically represented and detailed in tables, substantiates the effectiveness of this ML-centric approach. Service compositions managed by ML-Heal exhibit significantly faster recovery from faults, higher diagnostic accuracy, and maintain superior levels of availability when compared to traditional baseline strategies and simpler rule-based systems, all achieved with minimal requirement for human intervention.

The promising outcomes of this research pave the way for several exciting future directions. A key area for further investigation involves the exploration and integration of advanced predictive analytics and causal inference models at scale, enabling the system to anticipate failures even more proactively and understand fault propagation with greater precision [74]. Deploying and validating the ML-Heal framework in real-world, large-scale cloud environments, particularly those utilizing complex microservices architectures orchestrated by platforms like Kubernetes and Istio [75], will be crucial for assessing its practical viability and robustness under diverse operational loads and highly intricate failure scenarios. Further research into explainable AI (XAI) techniques [76] for the ML models used in diagnosis and planning would also enhance operator trust and facilitate easier debugging of the self-healing system itself. The inherent synergy between artificial intelligence and cloud orchestration technologies holds immense promise for constructing distributed applications that are not only more resilient but also possess advanced self-managing capabilities, adapting intelligently to the ever-changing conditions of dynamic cloud ecosystems [77].

ACKNOWLEDGMENT

The authors extend their appreciation to the Deanship of Scientific Research at Northern Border University, Arar, KSA for funding this research work through the project number NBU-FFR-2025-2119-07.

REFERENCES

[1] V. Kaushik, P. Bhardwaj, and K. Lohani, "Game of definitions—do the nist definitions of cloud service models need an update? a remark," in *Futuristic Trends in Networks and Computing Technologies: Select Proceedings of Fourth International Conference on FTNCT 2021*. Springer, 2022, pp. 653–666.

[2] D. C. Marinescu, *Cloud computing: theory and practice*. Morgan Kaufmann, 2022.

[3] F. Siqueira and J. G. Davis, "Service computing for industry 4.0: State of the art, challenges, and research opportunities," *ACM Computing Surveys (CSUR)*, vol. 54, no. 9, pp. 1–38, 2021.

[4] A. Ganje, "Microservices in organizations," *Journal of Software Engineering and Applications*, vol. 18, no. 2, pp. 76–86, 2025.

[5] S. Newman, *Building Microservices: Designing Fine-Grained Systems*. O'Reilly Media, 2015.

[6] P. Rajeswari and K. Jayashree, "Design of an optimal deep learning-based self-healing mechanism with failure prediction model for web services," *International Journal of e-Collaboration (IJeC)*, vol. 18, no. 1, pp. 1–15, 2022.

[7] V. V. R. Boda and J. Immaneni, "Keeping healthcare running smoothly: How sre is changing the game," *International Journal of Emerging Research in Engineering and Technology*, vol. 5, no. 3, pp. 43–51, 2024.

[8] B. Walsh, "Avoiding costly downtime–how msps can manage their networks," *Network Security*, vol. 2021, no. 2, pp. 17–19, 2021.

[9] S. S. Gill, A. Kumar, H. Singh, M. Singh, K. Kaur, M. Usman, and R. Buyya, "Quantum computing: A taxonomy, systematic review and future directions," *Software: Practice and Experience*, vol. 52, no. 1, pp. 66–114, 2022.

[10] Q. Al-Na'amneh, M. Aljawarneh, R. Hazaymih, L. Alzboon, D. A. Laila, and S. Albawaneh, "Trust evaluation enhancing security in the cloud market based on trust framework using metric parameter selection," in *Utilizing AI in Network and Mobile Security for Threat Detection and Prevention*. IGI Global Scientific Publishing, 2025, pp. 233–254.

[11] T. Wochner, "Part-time parliamentarians? evidence from outside earnings and parliamentary activities," *European Journal of Political Economy*, vol. 75, p. 102224, 2022.

[12] Y. Chen, M. Li, X. Zhu, K. Fang, Q. Ren, T. Guo, X. Chen, C. Li, Z. Zou, and Y. Deng, "An improved algorithm for practical byzantine fault tolerance to large-scale consortium chain," *Information Processing & Management*, vol. 59, no. 2, p. 102884, 2022.

[13] R. Barco, P. Lazaro, and P. Munoz, "A unified framework for self-healing in wireless networks," *IEEE Communications Magazine*, vol. 50, no. 12, pp. 134–142, 2012.

[14] M. Kirti, A. K. Maurya, and R. S. Yadav, "Fault-tolerance approaches for distributed and cloud computing environments: A systematic review, taxonomy and future directions," *Concurrency and Computation: Practice and Experience*, vol. 36, no. 13, p. e8081, 2024.

[15] A. Gupta and A. Gupta, "A vision and survey on autonomic computing," *International Journal of Multidisciplinary Innovative Research*, pp. 1–10, 2021.

[16] P. Dehraj and A. Sharma, "A review on architecture and models for autonomic software systems," *The Journal of Supercomputing*, vol. 77, no. 1, pp. 388–417, 2021.

[17] Q. Al-Na'amneh, R. Hazaymih, M. A. Almaiah, and L. Alzboon, "Secure cloud-marketplaces: A trust framework for evaluating security for client service providers," in *Utilizing AI in Network and Mobile Security for Threat Detection and Prevention*. IGI Global Scientific Publishing, 2025, pp. 255–280.

[18] H. Yang and Y. Kim, "Design and implementation of machine learning-based fault prediction system in cloud infrastructure," *Electronics*, vol. 11, no. 22, p. 3765, 2022.

[19] Q. Al-Na'amneh, A. N. Nasayreh, R. Al Mamlook, H. Gharaibeh, A. M. Alsheyab, and M. Almaiah, "Improving memory malware detection in machine learning with random forest-based feature selection," in *Risk Assessment and Countermeasures for Cybersecurity*. IGI Global, 2024, pp. 96–114.

[20] O. Gheibi, D. Weyns, and F. Quin, "Applying machine learning in self-adaptive systems: A systematic literature review," *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, vol. 15, no. 3, pp. 1–37, 2021.

[21] A. Sgueglia, A. Di Sorbo, C. A. Visaggio, and G. Canfora, "A systematic literature review of iot time series anomaly detection solutions," *Future Generation Computer Systems*, vol. 134, pp. 170–186, 2022.

[22] Y. Zhang, Y. Chen, J. Wang, and Z. Pan, "Unsupervised deep anomaly detection for multi-sensor time-series signals," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 2, pp. 2118–2132, 2021.

[23] M. Razian, M. Fathian, R. Bahsoon, A. N. Toosi, and R. Buyya, "Service composition in dynamic environments: A systematic review and future directions," *Journal of Systems and Software*, vol. 188, p. 111290, 2022.

[24] A. G. Barto, "Reinforcement learning: An introduction. by richard's sutton," *SIAM Rev*, vol. 6, no. 2, p. 423, 2021.

[25] D. Soni and N. Kumar, "Machine learning techniques in emerging cloud computing integrated paradigms: A survey and taxonomy," *Journal of Network and Computer Applications*, vol. 205, p. 103419, 2022.

[26] Q. Al-Na'amneh, M. Aljaidi, H. Gharaibeh, A. Nasayreh, R. E. Al Mamlook, S. Almatarneh, D. Alzu'bi, and A. S. Husien, "Feature selection for robust spoofing detection: A chi-square-based machine learning approach," in *2023 2nd International Engineering Conference on Electrical, Energy, and Artificial Intelligence (EICEEAI)*. IEEE, 2023, pp. 1–7.

[27] A. S. Jaradat, A. Nasayreh, Q. Al-Na'amneh, H. Gharaibeh, and R. E. Al Mamlook, "Genetic optimization techniques for enhancing web attacks classification in machine learning," in *2023 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*. IEEE, 2023, pp. 0130–0136.

[28] J. Alonso, L. Orue-Echevarria, E. Osaba, J. López Lobo, I. Martinez, J. Diaz de Arcaya, and I. Etxaniz, "Optimization and prediction techniques for self-healing and self-learning applications in a trustworthy cloud continuum," *Information*, vol. 12, no. 8, p. 308, 2021.

[29] M. Aljaidi, A. Alsarhan, D. Al-Fraihat, A. Al-Arjan, B. Igried, S. M. El-Salhi, M. Khalid, and Q. Al-Na'amneh, "Cybersecurity threats in the era of ai: Detection of phishing domains through classification rules," in *2023 2nd International Engineering Conference on Electrical, Energy, and Artificial Intelligence (EICEEAI)*. IEEE, 2023, pp. 1–6.

[30] P. Rauba, N. Seedat, K. Kacprzyk, and M. van der Schaar, "Self-healing machine learning: A framework for autonomous adaptation in real-world environments," *Advances in Neural Information Processing Systems*, vol. 37, pp. 42 225–42 267, 2024.

[31] R. Arora, A. Kumar, A. Soni, and A. Tiwari, "Ai-driven self-healing cloud systems: Enhancing reliability and reducing downtime through event-driven automation," *Preprints*, p. 2024081860, 2024.

[32] R. K. Vankayalapati and C. Pandugula, "Ai-powered self-healing cloud infrastructures: A paradigm for autonomous fault recovery," *Migration Letters*, vol. 19, no. 6, pp. 1173–1187, 2022.

[33] H. Xiao, K. Yi, R. Peng, and G. Kou, "Reliability of a distributed computing system with performance sharing," *IEEE Transactions on Reliability*, vol. 71, no. 4, pp. 1555–1566, 2021.

[34] G. Vidal, "An asynchronous scheme for rollback recovery in message-passing concurrent programming languages," in *Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing*, 2024, pp. 1132–1139.

[35] S. Gupta and M. Sadoghi, "Blockchain transaction processing," *arXiv preprint arXiv:2107.11592*, 2021.

[36] A. Lewis-Pye and T. Roughgarden, "Byzantine generals in the permissionless setting," in *International Conference on Financial Cryptography and Data Security*. Springer, 2023, pp. 21–37.

[37] T. Distler, "Byzantine fault-tolerant state-machine replication from a systems perspective," *ACM Computing Surveys (CSUR)*, vol. 54, no. 1, pp. 1–38, 2021.

[38] B. K. Singh, M. Danish, T. Choudhury, and D. P. Sharma, "Autonomic resource management in a cloud-based infrastructure environment," *Autonomic Computing in Cloud Resource Management in Industry 4.0*, pp. 325–345, 2021.

[39] B. Guo, J. Yu, D. Yang, H. Leng, and B. Liao, "Energy-efficient database systems: A systematic survey," *ACM Computing Surveys*, vol. 55, no. 6, pp. 1–53, 2022.

[40] E. Pimentel, W. Pereira, P. H. M. Maia, M. I. Cortés *et al.*, "Self-adaptive microservice-based systems-landscape and research opportunities," in *2021 International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)*. IEEE, 2021, pp. 167–178.

[41] T. Wong, M. Wagner, and C. Treude, "Self-adaptive systems: A systematic literature review across categories and domains," *Information and Software Technology*, vol. 148, p. 106934, 2022.

[42] A. D. Arndt, J. B. Ford, B. J. Babin, and V. Luong, "Collecting samples from online services: How to use screeners to improve data quality," *International Journal of Research in Marketing*, vol. 39, no. 1, pp. 117–133, 2022.

[43] C. S. Roldan, *React 17 Design Patterns and Best Practices: Design, build, and deploy production-ready web applications using industry-standard practices*. Packt Publishing Ltd, 2021.

[44] E. S. Kumar, R. Ramamoorthy, S. Kesavan, T. Shobha, S. Patil, and B. Vighneshwari, "Comparative study and analysis of cloud container technology," in *2024 11th International Conference on Computing for Sustainable Global Development (INDIACom)*. IEEE, 2024, pp. 1681–1686.

[45] N. Alfares, G. Kesidis, A. F. Baarzi, and A. Jain, "Cluster resource management for dynamic workloads by online optimization," *arXiv preprint arXiv:2207.04594*, 2022.

[46] S. Ahmad, I. Shakeel, S. Mehfuz, and J. Ahmad, "Deep learning models for cloud, edge, fog, and iot computing paradigms: Survey, recent advances, and future directions," *Computer Science Review*, vol. 49, p. 100568, 2023.

[47] M. H. Soleimani-Babakamali, R. Soleimani-Babakamali, R. Sarlo, M. F. Farghally, and I. Lourentzou, "On the effectiveness of dimensionality reduction for unsupervised structural health monitoring anomaly detection," *Mechanical Systems and Signal Processing*, vol. 187, p. 109910, 2023.

[48] O. Triebe, H. Hewamalage, P. Pilyugina, N. Laptev, C. Bergmeir, and R. Rajagopal, "Neuralprophet: Explainable forecasting at scale," *arXiv preprint arXiv:2111.15397*, 2021.

[49] H. Xu, G. Pang, Y. Wang, and Y. Wang, "Deep isolation forest for anomaly detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 12, pp. 12 591–12 604, 2023.

[50] V. Chernozhukov, D. Chetverikov, K. Kato, and Y. Koike, "High-dimensional data bootstrap," *Annual Review of Statistics and Its Application*, vol. 10, no. 1, pp. 427–449, 2023.

[51] X. Ma, J. Wu, S. Xue, J. Yang, C. Zhou, Q. Z. Sheng, H. Xiong, and L. Akoglu, "A comprehensive survey on graph anomaly detection with deep learning," *IEEE transactions on knowledge and data engineering*, vol. 35, no. 12, pp. 12 012–12 038, 2021.

[52] S. Bharany, S. Badotra, S. Sharma, S. Rani, M. Alazab, R. H. Jhaveri, and T. R. Gadekallu, "Energy efficient fault tolerance techniques in green cloud computing: A systematic survey and taxonomy," *Sustainable Energy Technologies and Assessments*, vol. 53, p. 102613, 2022.

[53] S. R. Katragadda, A. Tanikonda, S. R. Peddinti, and B. K. Pandey, "Machine learning-enhanced root cause analysis for accelerated incident resolution in complex systems," *Journal of Science & Technology*, vol. 2, no. 4, pp. 253–275, 2021.

[54] Z. Jin, Y. Chen, F. Leeb, L. Gresele, O. Kamal, Z. Lyu, K. Blin, F. Gonzalez Adauto, M. Kleiman-Weiner, M. Sachan *et al.*, "Cladder: Assessing causal reasoning in language models," *Advances in Neural Information Processing Systems*, vol. 36, pp. 31 038–31 065, 2023.

[55] R. Xin, P. Chen, and Z. Zhao, "Causalrca: Causal inference based precise fine-grained root cause localization for microservice applications," *Journal of Systems and Software*, vol. 203, p. 111724, 2023.

[56] Q. Cheng, A. Saha, W. Yang, C. Liu, D. Sahoo, and S. Hoi, "Logai: A library for log analytics and intelligence," *arXiv preprint arXiv:2301.13415*, 2023.

[57] Z. Chen, J. Liu, W. Gu, Y. Su, and M. R. Lyu, "Experience report: Deep learning-based system log analysis for anomaly detection," *arXiv preprint arXiv:2107.05908*, 2021.

[58] A. Feder, K. A. Keith, E. Manzoor, R. Pryzant, D. Sridhar, Z. Wood-Doughty, J. Eisenstein, J. Grimmer, R. Reichart, M. E. Roberts *et al.*, "Causal inference in natural language processing: Estimation, prediction, interpretation and beyond," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 1138–1158, 2022.

[59] Q. Ma, H. Li, and A. Thorstenson, "A big data-driven root cause analysis system: Application of machine learning in quality problem solving," *Computers & Industrial Engineering*, vol. 160, p. 107580, 2021.

[60] K. Zhu, J. Liang, J. Li, and C. Yi, "Coded distributed computing with predictive heterogeneous user demands: A learning auction approach," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 8, pp. 2426–2439, 2022.

[61] W. Zhang, D. Yang, H. Peng, W. Wu, W. Quan, H. Zhang, and X. Shen, "Deep reinforcement learning based resource management for dnn inference in industrial iot," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 8, pp. 7605–7618, 2021.

[62] A. MOUSA, "Trust management for context-aware composite services," Ph.D. dissertation, CONCORDIA UNIVERSITY MONTReal, QUebec, 2021.

[63] R. Van Dinter, B. Tekinerdogan, and C. Catal, "Predictive maintenance using digital twins: A systematic literature review," *Information and Software Technology*, vol. 151, p. 107008, 2022.

[64] Q. Cheng, D. Sahoo, A. Saha, W. Yang, C. Liu, G. Woo, M. Singh, S. Saverese, and S. C. Hoi, "Ai for it operations (aiops) on cloud platforms: Reviews, opportunities and challenges," *arXiv preprint arXiv:2304.04661*, 2023.

[65] Z. Zamanzadeh Darban, G. I. Webb, S. Pan, C. Aggarwal, and M. Salehi, "Deep learning for time series anomaly detection: A survey," *ACM Computing Surveys*, vol. 57, no. 1, pp. 1–42, 2024.

[66] R. E. Al Mamlook, M. Obeidat, N. Elgeberi, A. Nasayreh, A. A. Frefer, H. Gharaibeh, T. Gharaibeh, and Q. Al-Na'amneh, "Developing predictive maintenance for early warnings of machine failures using machine learning techniques in the fourth industrial era," in *2024 4th Interdisciplinary Conference on Electrics and Computer (INTCEC)*. IEEE, 2024, pp. 1–6.

[67] J. Bogner, J. Fritzsch, S. Wagner, and A. Zimmermann, "Industry practices and challenges for the evolvability assurance of microservices: An interview study and systematic grey literature review," *Empirical Software Engineering*, vol. 26, pp. 1–39, 2021.

[68] S. Gundawar, N. Kumar, P. Yash, A. K. Singh, M. Deepan, R. Subramani, B. Uma, G. Krishnapriya, B. Shivaprakash, and D. Venkataramana, "Multihead self-attention and lstm for spacecraft telemetry anomaly detection," in *International Advanced Computing Conference*. Springer, 2021, pp. 463–479.

[69] J. Wu, Z. Huang, Z. Hu, and C. Lv, "Toward human-in-the-loop ai: Enhancing deep reinforcement learning via real-time human guidance for autonomous driving," *Engineering*, 2023.

[70] I. Hafeez, "A systematic mapping study on self-adaptation sdn based iot networks," 2023.

[71] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[72] M. Towers, A. Kwiatkowski, J. Terry, J. U. Balis, G. De Cola, T. Deleu, M. Goulao, A. Kallinteris, M. Krimmel, A. KG *et al.*, "Gymnasium: A standard interface for reinforcement learning environments," *arXiv preprint arXiv:2407.17032*, 2024.

[73] A. Videla and J. J. Williams, *RabbitMQ in Action: Distributed Messaging for Everyone*. Manning Publications, 2012.

[74] L. Stone, "Causal inference in statistics: A primer," *Perspectives on Information Fusion*, vol. 3, no. 1, pp. 27–35, 2020.

[75] L. Calcote and Z. Butcher, *Istio: Up and running: Using a service mesh to connect, secure, control, and observe*. O'Reilly Media, 2019.

[76] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, F. Fdez-Navarro, A. Pastor-Lopez *et al.*, "Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai," *Information Fusion*, vol. 58, pp. 82–115, 2020.

[77] N. Agrawal, "Autonomic cloud computing based management and security solutions: State-of-the-art, challenges, and opportunities," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 12, p. e4349, 2021.

Exploring the Factors Influencing School Dropout: A Logit Model Analysis

Noaman LAKCHOUCH, Lamarti Sefian MOHAMMED, Mustapha KHALFOUNI
Abdelmalek Essaadi University, Tetouan, Morocco

***Abstract*—School dropout negatively impacts a country's development index, with numerous factors contributing to this complex phenomenon. To investigate the factors associated with school dropout in a specific region of Morocco, a cross-sectional study was conducted, encompassing a weighted sample of 274 junior secondary education students. Data collection was facilitated through a questionnaire administered to school directors. The data processing involved two main stages: preparation and modeling. The modeling phase employed a binary logistic regression model, focusing on the student's dropout status as the dependent variable. The study's findings highlighted several significant factors associated with school dropout: academic performance (as indicated by exam grades), the student's age and gender, and the availability of school transportation services that encourage students to continue their studies. Additionally, while class size also played a significant role, its impact was deemed less critical compared to the other factors identified. These results underscore that school dropout is influenced by a multitude of factors, suggesting the need for targeted interventions to prevent dropout and foster academic success, particularly among female students.**

Keywords*—*School dropout; development index; Morocco; cross-sectional study; logistic regression; academic performance; exam grades; age; gender; school transportation; academic success

I. Introduction

School dropout is a phenomenon that affects many countries around the world. It is defined as the interruption of studies before reaching higher education levels ([1], p. 26). This complex and multifactorial phenomenon is influenced by a set of factors such as poverty, learning difficulties, disinterest in school, school violence, discrimination, distance from school, among others [4]. School dropout has significant consequences for both individuals and society, leading to difficulties in professional integration, health problems, challenges in social and cultural development, and negatively impacting economic growth and social development [4]. According to the data from the Ministry of National Education, the school dropout rate in Morocco has slightly decreased over the past three years. In 2020 to 2021, it was 5.3%, with more than 331,000 students, girls and boys, leaving school. In 2021 to 2022, this rate remained at 5.3%, with 334,664 dropouts recorded. In 2022 to 2023, it slightly decreased to 5.2%, with 327,271 dropouts recorded [2]. This trend is also observed in secondary education in Morocco, where the dropout rate decreased from 2.9% in 2020-2021, with over 114000 students, to 2.8% in 2022 to 2023, with 112,925 dropouts recorded. For a geographical area located in the north of Morocco, there is a slight decrease in the dropout rate among secondary school students, dropping from 3.0% in 2020 to 2021, to 2.8% in 2022 to 2023, with a decrease of one-tenth of a percentage point (0.1%) per year [2]. Despite these encouraging results, many efforts are still needed to improve student retention in the education system. In fact, 90.4% of students enrolled in the first year of primary school reach the end of primary school (with repetition), and 71% of them manage to complete secondary education, compared to 90% in the United States, for example [3]. To better understand the causes of this phenomenon, we rely on personal data from 300 secondary school students, collected through a questionnaire distributed in 10 establishments in the study area. In doing so, we aim to assess the risk of school dropout and identify the influence of students' personal and academic characteristics on their intentions to leave school. These two objectives are achieved by answering the following question: What are the personal and socio-economic factors contributing to school dropout in secondary education?

We conducted our study in several stages: a literature review allowed us to examine the main previous studies. In a section dedicated to tools and methods, we used a logistic model to identify the factors significantly influencing the decision to drop out of school before completing the cycle. In a section devoted to presenting the results of our statistical modeling, we presented the findings of our study followed by a discussion. In conclusion, we provided the final remarks of our study.

II. Literature Review

School dropout is a major issue with significant consequences for individuals, families, and societies. In this study, it is defined as the interruption of education before obtaining a secondary school diploma in Morocco, more specifically within the study area during the 2022 to 2023 academic year. At the global level, the phenomenon remains alarming. According to a UNESCO report (2020), 262 million school-aged children and adolescents were out of school, of which 123 million were girls [4].

Numerous international and national studies have examined the factors contributing to dropout. For instance, [5] highlighted the strong link between school failure and social exclusion. His comparative study between France and the United Kingdom showed that school exclusion is closely related to social exclusion, each reinforcing the other. Similarly, [6] emphasized the multifactorial nature of the phenomenon, pointing to economic, family, school-related, and personal determinants.

In Croatia, [7] identified four categories of dropouts: low-performing students, "silent" disengaged dropouts, maladjusted dropouts with emotional difficulties, and "stressed"

dropouts overwhelmed by academic demands. This typology illustrates the diversity of dropout pathways and the interdependence of risk factors.

In the United States, [8], through a longitudinal study of 6,330 ninth-grade students, demonstrated the importance of both individual and contextual characteristics. His findings showed that gender (boys being more at risk), academic performance, part-time employment, family background, and the socio-economic environment of schools were significant determinants of dropout. Similarly, [9] found that academic difficulties, low self-efficacy, and behavioral issues significantly increase the risk of dropout, especially when combined with contextual factors such as poverty and low parental education.

From a gender perspective, [10] studied early school leaving among girls in Spain, showing that stereotypes, discrimination, violence, and the lack of institutional or family support play a critical role. This highlights how gender acts as a transversal factor that amplifies the effects of other social and school-related determinants.

In Morocco, several studies have also examined this phenomenon. In [11], the authors found that male students from disadvantaged families and those with grade repetitions were more likely to leave school. Similarly, [12], [13] categorized the determinants of dropout into individual, socio-economic, educational, and family-related factors. They noted that socio-economic reasons tend to affect boys more, whereas educational difficulties are more significant for girls. More recent research [14] has further stressed the importance of institutional factors (such as school quality and discipline) and socio-economic conditions (family income, rural environment).

Taken together, these studies confirm that school dropout is a complex and multidimensional phenomenon, shaped by the interaction of individual factors (academic performance, age, gender, risky behaviors, and mental health), family factors (family structure, parenting practices, parental education, socio-economic background), and school-related or social factors (class size, school climate, educational policies, and socio-economic environment). However, several limitations can be identified. Few Moroccan studies employ robust quantitative methods capable of evaluating the relative effects of different variables. Support services such as school transport, canteens, or boarding facilities are seldom analyzed, even though they are critical for keeping students in school, particularly in rural areas. The gender dimension, while mentioned, is often addressed superficially, without deeper exploration of the mechanisms making girls more vulnerable. Finally, many studies are constrained by small samples or rely exclusively on qualitative approaches, which makes it difficult to generalize the findings.

It is within this context that the present study positions itself. By applying a binary logit model to a weighted sample of junior secondary school students in northern Morocco, it provides a rigorous statistical measure of the determinants of dropout. It also highlights the crucial role of school transportation, a factor rarely addressed in previous studies, confirms the differentiated impact of gender, and offers recent empirical evidence to inform educational policies aimed at preventing dropout in Morocco.

III. Materials and Methods

A. Sources of Data

In collaboration with the educational authorities of the study area, a simple random survey (SRS) was conducted to select ten junior secondary schools. Principals from these selected schools were then requested to complete a questionnaire designed to collect data on students who either dropped out or completed their secondary education during the 2022 to 2023 academic year. This effort resulted in the collection of 274 records, representing individual students or observations.

The questionnaire was divided into two primary sections. The first section gathered details about the participating schools, including their location, staff composition, student enrollment numbers, and infrastructure facilities. The second section, designed as a table, collected individualized information on the students selected for the study. This data encompassed sociodemographic aspects such as age, gender, family background, and academic history (including dates of enrollment and withdrawal, instances of grade repetition, and average academic scores). Additionally, it collected information on participation in social assistance programs like boarding facilities and school transportation services. A set of variables was then derived from this collected information to streamline the statistical analysis process, which are outlined in Table I.

B. Sample Weighting (Data Preprocessing)

The student selection process was implemented through a mixed-method survey approach that used both stratified and cluster sampling techniques. Initially, ten schools were randomly chosen, each with an equal chance of selection from a total of 94 schools.

Subsequently, within each of these schools, every student who dropped out during the academic year 2022 to 23 was included in the study. Additionally, a comparable number of third-year students were randomly selected. Thus, the calculation of individual weights for the students proceeded are as described:

Computation of Sampling Weights [15], [16]:

If the student is abandoned:

$$\omega_i = \frac{1}{prob(i \in S_0)}$$

Here, S_0 denotes the sample of selected institutions (schools).

The idea is that the probability of selecting a dropout student depends solely on the probability that their institution was chosen in the sample. Thus, each dropout student observed in the sample represents the inverse of this probability.

For example:

10 schools are chosen out of 94 $\rightarrow$ probability $= \frac{10}{94}$.

Therefore, the weight is $(\frac{94}{10}) = 9.4$.

This means that each dropout student represents 9.4 students in the population.

TABLE I. DESCRIPTION OF VARIABLES

Variable	Description
Statut	A binary categorical variable takes the value 1 if the student leaves school and 0 if not
Std.Tchr	Supervision rate, this is the number of students per teacher (StudentTeacher)
Std.Admin	Number of students per administrator (students/administrator)
Std.Serv	The number of students per support service member
Acad.Rat	The proportion of academy staff to the total number of teachers
Std.Cl	Average number of students per class (rate=students/class)
Std.Cl.Rm	Average number of students per classroom (rate=students/classroom)
Std.L	The students' academic level
Space.H	Number of classrooms at the school
Sprt.F	The existence or not of the sports field at the school (1 if yes 0 if no)
Water	The existence of the water area at the school (1 if yes, 0 if no)
Electr	The existence of the electricity area of school (1 if yes, 0 if no)
Sanit.Serv	The existence of sanitation services (1 if yes, 0 if no)
Age	Age of student
Gender	Gender of student
Rep.Nb	Number of repeats of the student since his/her first access to school
Score	Average mark of the student in three subjects (Arabic and French language, Maths)
Dist.S.H	The distance covered by the student to arrive at the school
Partl.Sit	Parental situation (two-parent (2) or one-parent (1) household)
Luch.Sch	The student is a beneficiary of the school lunch program (1 if yes, 0 if no).
Std.Internat	A derived categorical variable that takes the value 2 for a student who is a beneficiary of the boarding school, 1 if the boarding school exists but the student is not a resident and vice versa, and 0 if there is no boarding school and not a beneficiary.
Trspt.B	The student benefits from the school transportation program (1 if yes, 0 if no)
Tayssir.B	the student benefits from the Teyssier program (1 if yes, 0 if no)

If the student has completed their course

$$\omega_i = \frac{1}{prob(i \in S_0)} \times \frac{1}{prob(i \in S_E)}$$

where, S_E is the sample of students selected within the institution.

Therefore, the probability of selecting a dropout student is simply the probability of choosing their institution; consequently, their probability of belonging to the sample is the same as that of selecting the institution.

In practice, if a student has dropped out, their weight is $\omega_i = (\frac{10}{94})^{-1} = \frac{94}{10} = 9.4$, meaning each selected dropout student in the sample represents 9.4 students. If the student has completed their course, we have $\omega_i = \frac{94}{10} \times \frac{N_e}{n_e}$, where N_e is the number of students in the 3rd year of junior secondary education in institution e and n_e is the number of selected students from the same institution.

C. Methods

1) Model specification: Our aim is to predict whether a student is likely to drop out of school by examining both their personal characteristics and the features of their school environment. Thus, school dropout is viewed as a binary outcome, where a student either discontinues their education prematurely or completes their secondary schooling. This scenario satisfies the prerequisites for binary classification, enabling the use of the binary logistic regression model [17].

Binary logistic regression (LR) is a statistical technique used to forecast the likelihood of a binary outcome, based on a set of independent variables. This outcome is binary on nature, signifying it can assume one of two possible values, such as "yes" or "no", or "success" or "failure" [18].

The LR model is presented as a logistic function, which is a sigmoid function that takes a real number as input and outputs a value between 0 and 1. The logistic function is defined as:

$$Logistic(x) = \frac{1}{1 + e^{-k(x+x_0)}}$$

where, x is a real number. The logistic function (see Fig. 1) has the properties illustrated in the following graph:

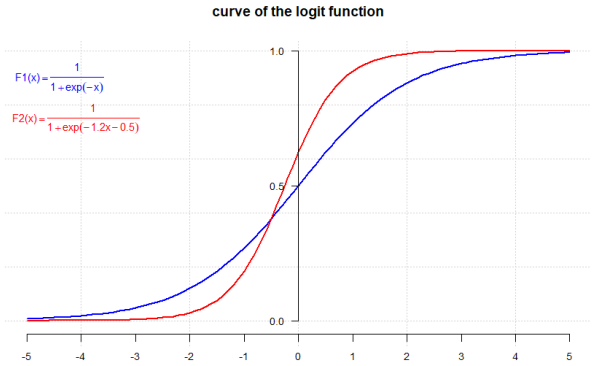


Fig. 1. Graphical representation of the logistics function.

The logistic function is used in the logistic regression model to transform the linear combination of the predictor variables into a probability (Boateng, et al. 2019) [17].

The logistic regression model is defined as:

$$P(Y = 1|X) = Logistic(\beta_0 + \beta_1 X1 + \beta_2 X2 + ... + \beta_3 Xp)$$

where, Y is the binary outcome variable, X1, X2, ..., Xp are the predictor variables, β_0 is the intercept, and β_1, β_2, ..., β_p are the coefficients.

Here is an example of a logistic regression model that predicts the probability of a student passing an exam based on their GPA and the number of hours they studied and gender [Eq. (1)].

$$P(\text{Pass}|\text{GPA, Hours Studied, Gender}) = Logit(\beta_0 + \beta_1 \text{Score} + \beta_2 \text{Hours Studied} + \beta_3 \text{Gender}) \quad (1)$$

Score: The student's academic performance, Hours Studied: The number of hours they studied, Gender: The student's gender, β_0: The intercept and β_1, β_2, and β_3: The coefficients.

The logistic regression model can be used to predict the probability of a student passing the exam for any given Score, number of hours studied and Gender. The model can also be used to identify the factors that are most important for predicting whether a student will pass the exam [19].

2) Model selection: According to the analysis of the AIC index, we can eliminate a certain number of variables and retain the following variables: Std.Cl, Std.Srv, Age.Std, Gender, Rep.Nb, Score.Av, Std.L, and Trspt.B. The AIC value decreases from 63.76 for a saturated (complete) model to 61.57 for a model containing only the mentioned variables [Eq. (2)].

3) Model Validation: The VIF [1] analysis allows us to conclude that there is no collinearity among the explanatory variables (all VIFs are below 5).

Furthermore, the ROC curve (see Fig. 2) analysis on the test data confirms that this model is excellent for predicting school dropout in a student, with an AUC of 0.87.

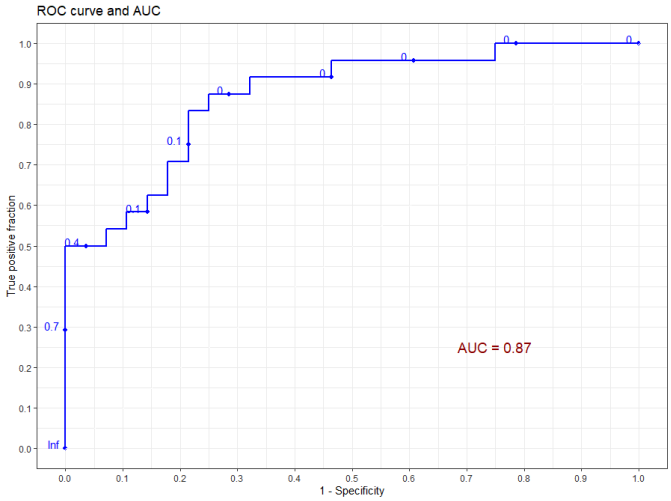


Fig. 2. ROC curve.

Still, on the test data, we perform the Hosmer-Lemeshow test [2] [19]. The test allows us to say that the model is adequate $(Statistic = 8.736, df = 8, p - value = 0.365)$.

[1]The Variance Inflation Factor (VIF) is a measure used to detect the presence of multicollinearity in the model such as:

$VIF = 1$: Indicates that there is no correlation between the independent variable in question and the other independent variables in the model. In other words, there is no multicollinearity.

$1 < VIF < 5$: VIF values in this range are generally considered acceptable. There is some correlation, but it is not high enough to cause serious multicollinearity problems.

$VIF \geq 5$: A VIF of 5 or more indicates multicollinearity that may be concerning and requires attention. This suggests that the independent variable is strongly correlated with other explanatory variables in the model.

[2]The Hosmer-Lemeshow test is a statistical test used to evaluate the goodness of fit for logistic regression models. Specifically, it tests whether the observed event rates match expected event rates in subgroups of the model population. The null hypothesis for the Hosmer-Lemeshow test is that there is no difference between the observed and expected frequencies of the outcome variable

4) Model estimation: After selecting and validating the model on the test data, we can keep the following logistic regression model:

$$P(\text{Statut} = 1|X) = \text{Logit}(\beta_0 + \beta_1 \text{Std.Cl} + \beta_2 \text{Std.Srv} + \beta_3 \text{Age.Std} + \beta_4 \text{Gender} + \beta_5 \text{Rep.Nb} + \beta_6 \text{Score.Av} + \beta_7 \text{Std.L} + \beta_8 \text{Trspt.B}) \quad (2)$$

TABLE II. THE ESTIMATION OF THE FINAL LOGIT MODEL

Characteristic	Beta[1]	Sig.[2]	OR[3]	95% CI (OR)	Sig.	VIF[4]
Std.Cl	0.203	0.062	1.23	0.99–1.52	0.065	4.7
Std.Srv	0.004	0.054	1.00	1.00–1.01	0.054	3.8
Age.Std	0.497	0.019	1.64	1.09–2.48	0.018	3.2
Gender					0.011	1.3
Girl	Ref[5]					
Boy	-0.117	0.012	0.33	0.14–0.78	0.012	
Rep.Nb	0.561	0.023	1.75	1.08–2.84	0.001	2.6
Score.Av	-0.601		0.56	0.45–0.68	0.001	1.7
Std.L	-1.400	0.001	0.25	0.10–0.59	0.001	1.8
Trspt.B		0.019			0.019	1.9
No	Ref					
Yes	-1.432	0.020	0.24	0.07–0.80	0.020	

IV. RESULTS

The results from Table II allows us to conclude that the number of students per service, the age and gender of the student, the number of grade repetitions, the average score in the main subjects (Math, French, and Arabic), the student's academic level, and the use of school transportation are factors that significantly influence school dropout differently. Thus, the average number of students per class has a positive influence $(beta = 0.213, p - value = 0.061)$, as well as the number of students per service and age, which also have positive effects on dropout. Specifically, an additional year in age increases the risk of leaving school by 1.43 times $(OR = 1.43, p - value = 0.042)$, and each additional repetition of a grade increases the chance of leaving school by 1.75 times $(OR = 2.13, p - value < 0.001)$. On the other hand, some factors have negative and significant effects, meaning they decrease the risk of dropping out of school. For instance, being a boy makes the chance of not dropping out of school 1/0.38=2.6 times higher than for a girl, indicating that girls are more likely to leave school than boys. School dropout is influenced by various factors, but some are more important than others. The average score in main subjects is the most crucial factor, with a weight of 5.78. The student's academic level is also important, with a weight of 3.19. Age, gender, the existence of school transportation services, and the number of grade repetitions have moderate importance, with weights ranging from 2.33 to 2.54. The number of students per class does not have a significant impact on the risk of school dropout (see Table III).

V. DISCUSSION AND IMPLICATIONS

As mentioned earlier, the objective of this analysis was to identify the factors that significantly influence school dropout

TABLE III. THE IMPORTANCE OF FACTORS

Factor (Variable)	Weight
Std.Cl	1.845
Std.Srv	1.929
Age.Std	2.372
Gender	2.545
Rep.Nb	2.289
Score.Av	5.788
Std.L	3.191
Trspt.B	2.338

in junior secondary education. Based on a weighted sample of 274 observations, our findings confirm several results from previous studies while also offering new insights specific to the Moroccan context.

Regarding student characteristics, academic performance emerges as the most critical determinant, consistent with the findings of [8], [9]. Students with poor grades or repeated years face a substantially higher risk of dropping out, highlighting the importance of academic support and remedial programs. Age and gender also play a role, though to a lesser extent. In line with [10], our study shows that girls are more likely to drop out than boys when academic performance is weak, suggesting that gender-related vulnerabilities may exacerbate the effect of poor performance. This underscores the need for gender-sensitive policies aimed at keeping girls in school, particularly in rural and disadvantaged contexts.

With respect to school-related factors, our results provide a more nuanced perspective. While smaller class sizes are associated with lower dropout risk, their effect appears modest compared to student performance and support services. This aligns with the mixed evidence reported in earlier research [6]. More importantly, our analysis identifies school transportation as a significant factor in student retention—an aspect that is often overlooked in international studies but is particularly relevant in Morocco, where geographical distance and lack of infrastructure remain barriers to education. By contrast, the provision of boarding schools with catering services did not show a significant impact, suggesting that not all forms of institutional support contribute equally to reducing dropout.

Our analysis also reveals certain limitations. First, the study relied on cross-sectional survey data, which restricts the ability to capture the dynamics of dropout over time. Second, due to institutional constraints, family-related variables such as parental education, household income, and family structure could not be fully incorporated, even though prior research suggests that they are highly relevant [11], [12]. Addressing these limitations in future studies would allow for a more comprehensive understanding of the dropout phenomenon.

Future research should therefore build on this work by integrating family and community-level characteristics, employing longitudinal data to assess causal relationships, and exploring interaction effects between academic performance, gender, and socio-economic status. Moreover, additional attention could be given for evaluating the effectiveness of targeted interventions, such as transportation subsidies, remedial classes, or gender-sensitive programs, to determine which policies are most effective in reducing dropout rates in the Moroccan context.

VI. CONCLUSION

In this study, we analyzed the determinants of school dropout using weighted survey data from junior secondary education students in Morocco. The results show that school dropout is primarily driven by academic performance, with interacting effects of age and gender. In particular, girls with weak academic achievement are more likely to leave school prematurely. Moreover, our analysis highlights the importance of social support services, such as school transportation, which significantly reduce dropout risk by improving accessibility and retention. These findings provide valuable insights for policymakers seeking to design effective interventions against school dropout.

Our results suggest that educational support programs are essential to help students enhance their academic performance, with particular attention to vulnerable groups such as girls and students in rural areas. At the same time, logistical support measures, notably transportation, should be prioritized as effective levers to ensure equity in access to education.

This study nevertheless presents some limitations. It relies on cross-sectional survey data, which, although weighted to reflect the population, cannot capture the longitudinal dynamics of dropout nor the full range of family-level factors such as parental education, income, and household structure. Consequently, the results should be interpreted with caution and cannot be generalized to the entire population without further validation.

Future research should extend this work by incorporating longitudinal data, exploring family and community-level determinants, and assessing the effectiveness of targeted interventions. Furthermore, emerging methodologies such as machine learning and deep learning models offer promising avenues to predict dropout risk with greater precision, identify complex patterns of interactions between variables, and provide early warning systems for students at risk. Combining statistical modeling with artificial intelligence approaches could therefore enhance both the accuracy of dropout prediction and the design of preventive strategies tailored to the Moroccan educational context.

ACKNOWLEDGMENT

The authors would like to thank the school directors and students who participated in the survey, as well as the educational authorities who facilitated access to the data. Special thanks are also extended to colleagues and reviewers for their constructive comments, which helped improve the quality of this work. Any remaining errors are solely the responsibility of the authors.

REFERENCES

[1] La Bouture, Les lycéens décrocheurs : de l'impasse aux chemins de traverse, Chronique sociale, Librairie Mollat, 15 rue Vital-Carles,33 080 Bordeaux Cedex, 1998.

[2] MEN, Annuaire statistique de l'éducation, Ministère de l'Education Nationale, 2022.

[3] W.Bank, Lower secondary completion rate, World Bank, Website, 2022. https://data.worldbank.org/indicator/SE.SEC.CMPT.LO.ZS?locations=MA

[4] UNESCO, Rapport mondial sur l'éducation 2020 : par amour de l'apprentissage, Paris, FRANCE, 2020.

[5] Danielle Zay, Prévenir l'exclusion scolaire et sociale des jeunes. Une approche franco-britannique, Paris, PUF, 2005.

[6] Muller F., & Rocher G., L'abandon scolaire : un phénomène complexe et multifactoriel, Éducation et formation, 77(1), 1-16, 2014.

[7] Ogresta J., & Rezo I., Why do we drop out? Typology of dropping out of high school, Youth & Society, 53(6), 934-954, 2021.

[8] Zvoch, K., Freshman year dropouts: Interactions between student and school characteristics and student dropout status, Journal of Education for Students Placed at Risk, 11(1), 97-117, 2006.

[9] Gençtanır Ö., & Ercan, E., Prediction of school dropout among Turkish high school students: A multilevel analysis, Educational Studies, 58(1), 1-17, 2022.

[10] Rodríguez-Gómez, M. J., School dropout for girls: An intersectional perspective, Gender & Education, 35(2), 202-218, 2023.

[11] BIYOUDA S, et al., Les déterminants sociodémographiques et scolaires du risque de décrochage scolaire chez des collégiens marocains, Revue Française d'Economie et de Gestion, 2(3), 252-266, 2021.

[12] Ait Hammou M., & Bettach A., Causes de l'abandon scolaire au Maroc : une étude exploratoire, Revue Marocaine des Sciences de l'Education, 28(1), 117-134, 2021.

[13] Ait Hammou M., & Bettach A., L'abandon scolaire au Maroc : analyse des facteurs socio-économiques et institutionnels, Economie et Humanisme, (535), 55-64, 2022.

[14] Vinciguerra, A. et al., Les déterminants du décrochage dans l'enseignement secondaire : une revue de littérature, Revue française de pédagogie, (202), 111-130, 2020.

[15] Lohr, S. L., *Sampling: Design and Analysis*, 2nd ed., Chapman and Hall/CRC, New York, 2019.

[16] Bethlehem, J., *Applied Survey Methods: A Statistical Perspective*, Wiley, Hoboken, NJ, 2009.

[17] Boateng, E. and Abaye, D., A Review of the Logistic Regression Model with Emphasis on Medical Research, Journal of Data Analysis and Information Processing, (7), 190-207, 2019.

[18] Hyeoun-Ae Park, An Introduction to Logistic Regression: From Basic Concepts to Interpretation with Particular Attention to Nursing Domain, Journal of Korean Academy of Nursing, 43(2), 154-164, 2013.

[19] Hosmer D.W. and Lemeshow S., Applied Logistic Regression, John Wiley & Sons, New York, 2nd ed., 2000.

Securing the Healthcare Supply Chain Using Blockchain-Enabled Smart Contracts

Muhammad Saad[1], Hamail Ashraf[2], Muhammad Saffi Ullah Khan[3],
Muhammad Awais Javed[4], Ahmad Naseem Alvi[5], Ahmed Alfakeeh[6]*
Department of Electrical Engineering, COMSATS University Islamabad, 45550, Islamabad, Pakistan[1,2,3,4,5]
Department of Information Systems, King Abdul Aziz University, Jeddah, Saudi Arabia[6]

***Abstract*—The blockchain technology is an innovative tool that has shown its effectiveness in many sectors such as health-care and many are likely to experience a revolutionary transformation. In healthcare, the blockchain functions like a distributed network which is constantly updated with records ensuring that you cannot eliminate or alter them without the consensus to do so. In other words, blockchain-based smart contracts will enable clients to do transactions without having to rely on intermediaries and thus it will be more reliable. Smart contracts regulate and constitute numerous activity and dealings of stakeholders, automating processes, augmenting visibility, maximizing productivity, and standing against the clock. Healthcare supply chains may be exploited by blockchain technology that addresses pain points such as connectivity, traceability, as well as fighting counterfeit medicines. The purpose of our analysis was to advance traceability between the healthcare supply chain elements, such as the transportation of pharmaceuticals or medical equipment. And for its more practical implementation we have worked on minimizing the cost of smart contracts being deployed on the blockchain for the health care logistic system.**

***Keywords*—Blockchain; smart contracts; supply chain**

I. Introduction

With the advancement in technology and everything being digitalized, the supply chain management has become difficult to run by the old methods. With the beginning of a new era and digitalized market, it has become difficult to keep record of the supply chain system. From keeping record of raw material, manufacturing, transportation of goods to the market and keeping record of whether the items are damaged, expired, delivered, or lost, the need for a digitalized and secure system has emerged. Many difficulties are faced during transportation of goods like food, medicines, machineries, electronics, and almost all the items that can be damaged during long-distance transportation. This leads to complexity and ambiguity between the buyer and the supplier. So, the overall system becomes inefficient. COVID-19 is a major example [1], [2], [3] of why we need to move on to a new digital supply chain system that can provide more efficiency and transparency in the whole supply chain market [4], [5], [6].

We can exclude these problems and make a more transparent system by shifting towards the smart contract between the supplier and the buyer. These smart contracts will not just provide transparency but also will execute automatically, eliminating the middleman. Smart contracts are capable of keeping records safe on a blockchain system. These smart contracts can help in eliminating the constraints between the customer and the manufacturer. Records are saved in a ledger in a blockchain consisting of secured blocks that can eventually reduce the payment issues, increase the efficiency of the supply chain system, and reduce the problems being faced during transportation in the healthcare system. All the data will be stored on the supply chain in different blocks, which will also eliminate the risk of data being on a centralized unit [7], [8], [9], [10].

Major problems in a healthcare supply chain system are keeping track record of in-stock items, shortage of items, expiry of medicines, and managing the required temperature of the medicine. These problems must be dealt with; otherwise, the system becomes inefficient. For this, open-source blockchains are used in which smart contracts are translated and then deployed. Transaction of any agreement will only be done once both parties have fulfilled their respective jobs. This reduces the ambiguity in the supply chain of the healthcare system. For instance, if a customer receives any damaged or expired medicine, then his transaction will not be completed unless or until the agreements of the smart contracts are fulfilled.

After moving towards a digital smart contract system which is secure and reliable, it is important to keep account of the cost of using a smart contract-enabled supply chain system. For that, work on reducing the gas amount is necessary to make it feasible to be deployed in the healthcare supply chain system.

Concluding this, a smart contract-enabled healthcare supply chain system is necessary to fill the flaws in the old supply chain system, such as keeping the record safe, ensuring efficiency, having a decentralized system, and having transparency among the buyer and the manufacturer by eliminating the middleman from this whole process.

A. What is Blockchain?

The term *blockchain* was first introduced in 1991 by Stuart Haber and W. Scott Stornetta, describing the concept of linked blocks secured by cryptographic codes. Bitcoin was the first real-world application to implement blockchain technology, with its public version 1.0 launched in 2009. Later, the second generation of blockchain, referred to as version 2.0, emerged around 2013–2014 and gained practical use with the launch of Ethereum and its smart contracts in 2015. More recently, industries have adopted version 3.0 of blockchain, aiming to develop custom solutions suited to their own needs.

*Corresponding authors.

Blockchain is essentially a decentralized platform that enables secure and transparent sharing of information. It operates as a distributed digital ledger—a modern equivalent of traditional account books—where transactions are recorded. This ledger is maintained across a peer-to-peer network, making the data nearly tamper-proof. Each transaction requires digital signatures for authentication, thus minimizing the risk of fraud. The decentralized and immutable nature of blockchain makes the stored data highly secure and reliable.

B. Blocks in Blockchain

Blockchain is a decentralized and distributed digital ledger that records transactions across multiple systems, making them secure and transparent. At the core of this technology are *blocks*, which are digital containers used to store data. Each block includes a list of transactions, a timestamp, and a unique cryptographic hash as well as the hash of the previous block.

These blocks are connected sequentially, forming a chain—hence the name *blockchain*. The security of blockchain arises from the interlinking of blocks through cryptographic hashes, which ensures that each block references its predecessor. When a new transaction occurs, it is grouped with others into a block. Once the block is full, it is added to the blockchain and linked to the previous block using a hash function.

The hashing process uses SHA-256 (Secure Hash Algorithm 256-bit), a cryptographic function that generates a fixed-length, 256-bit signature from input data. This hash acts as a digital fingerprint, providing a unique identifier for each block. Any modification in a block will change its hash, thus invalidating the chain if the tampering is not corrected throughout.

This immutable nature of hash chaining ensures that data cannot be altered once added, making blockchain systems tamper-resistant. During the formation of a block, the system computes the hash of the block's data. This hash not only acts as the block's unique ID but also serves as a link to its predecessor.

Miners use SHA-256 to construct a proof-of-work, solving a computational puzzle to find a valid random integer (nonce). This process adds security and consensus to the network. Additionally, users interact with the blockchain using public and private cryptographic keys, ensuring secure communication and transaction validation.

C. Types of Blockchain

1) Public blockchain: Public blockchains are also called permission-less blockchains. Anyone can participate in the validation of transactions and data. Public blockchains are suitable for networks that have strict transparency requirements. Users can view both recent and historical records and can participate in mining. Bitcoin and Ethereum are examples of public blockchains.

2) Private blockchain: Private blockchains are used within specific organizations for the validation of transactions and data. They are more private and secure than public blockchains. Although they maintain a peer-to-peer connection and decentralized structure, they operate on a smaller scale. Only authorized parties can access this blockchain and participate in the network. Hyperledger and Corda are examples of private blockchains.

3) Hybrid blockchain: Hybrid blockchains combine characteristics of both public and private blockchains. Businesses that require both transparency and privacy often prefer this model. By using hybrid blockchains, organizations can establish both public and private systems, control which blockchain data is publicly accessible, and define who has access to specific data.

4) Consortium blockchain: A consortium blockchain is a permissioned blockchain where access decisions are controlled by a pre-selected group of nodes. It is commonly used by groups of organizations that aim to collaborate and share data in a decentralized manner while retaining a level of control over the network. Participants in a consortium blockchain typically know and trust each other. This model is well-suited for industries where multiple organizations need to collaborate while preserving data privacy and integrity.

D. Hashing Algorithm

A cryptographic security algorithm known as Secure Hash Algorithm 256-bit (SHA-256) is used in blockchain systems. This algorithm is secure and computationally difficult to reverse. SHA-256 generates hash values that are unique and irreversible, and each block in the blockchain is assigned such a hash value. Being a 256-bit value, it ensures that no two different blocks can have the same hash. Blockchain encryption is achieved by applying this algorithm, which involves a series of mathematical operations that transform input data into a fixed-length encrypted output. This process guarantees the integrity and security of the data within the blockchain.

E. Smart Contract and Use Cases

Smart contracts are digital agreements that can automatically execute predefined "if/else" conditions. These contracts are deployed on the blockchain and execute automatically when the encoded terms, defined by the participating parties, are met. Smart contracts enhance transparency and reliability by eliminating the need for intermediaries. Once a transaction is completed, it is irreversible. Depending on the type of blockchain, only authorized individuals may access the outcome of a transaction.

In supply chain systems, for example, a transaction is finalized only when the receiver confirms the items were delivered as specified in the smart contract. Otherwise, the transaction is halted until all conditions are satisfied. This enhances supply chain efficiency and reduces risk. Additionally, costs are minimized as smart contracts automate processes without requiring middlemen.

Smart contracts have several real-world applications:

- Financial services: Smart contracts enable decentralized finance (DeFi), automating trading and asset management without intermediaries. They also facilitate automated payments for organizations.
- Supply chain management: They track goods from raw material collection to delivery, ensuring quality control throughout the transportation process.

- Voting systems: Smart contracts help secure election processes, enhancing integrity and preventing manipulation.
- Social media and copyright: Content ownership and copyright enforcement can be managed through smart contracts to prevent unauthorized use of data.

F. Problem Statement

Traditional healthcare supply chain systems face challenges related to inefficiency and complexity. Tracking goods during transportation has become increasingly difficult. Common issues include maintaining inventory records, item shortages, medicine expiration, and temperature control requirements.

Ambiguity in manual record-keeping, corruption, and breaches of supply chain agreements have led to distrust between suppliers and customers. This highlights the need for a secure and digital smart contract-enabled system. Although smart contracts are partially deployed in this sector, their high deployment cost has hindered widespread adoption.

While these systems reduce overall service costs by eliminating intermediaries, the cost of deploying smart contracts on blockchain platforms remains high. Therefore, optimizing and reducing the gas fees associated with smart contract execution is essential for expanding this technology in the global healthcare supply chain.

II. Literature Review

In [11], the authors proposed a framework for Personal Health Records (PHR), highlighting the issue that a patient's personal health data is highly confidential and may be exposed to hospitals in emergencies or uncertain situations. To ensure security, the author introduced an emergency healthcare system that manages PHR using blockchain-enabled smart contracts with permissioned data access.

In [12], the authors proposed a mobile healthcare solution utilizing smart contracts on blockchain. This system facilitates rapid action in emergencies where a doctor must urgently assess a patient. Patients can transmit real-time data such as blood pressure or oxygen levels via IoT devices, and doctors can immediately respond. The entire process is executed through smart contracts on the blockchain, enhancing both security and speed.

In [13], the authors addressed ambiguities in the supply chain system, particularly within the cyber and agricultural sectors. The study proposed a blockchain-based solution to address inefficiencies and enhance traceability and transparency.

In [14], the researchers focused on the supply and demand issues of COVID-19 medical equipment. The study highlighted flaws in traditional, centralized supply chain systems, such as their vulnerability to failure and lack of reliability. To overcome these, a blockchain-enabled supply chain model was proposed, offering decentralization, enhanced security, and improved trust among stakeholders.

In [15], the researchers reviewed over 100 published articles on Electronic Health Records (EHR). They compared the security, cost, and privacy of traditional supply chain systems versus blockchain-based systems, demonstrating the superiority of blockchain in terms of data integrity and privacy protection.

In [16], a solution for an Internet-based healthcare system was presented. The system allows patient appointments to be managed remotely using a blockchain-integrated app. Only authorized users or patients can access the data, ensuring privacy and preserving the integrity of health records.

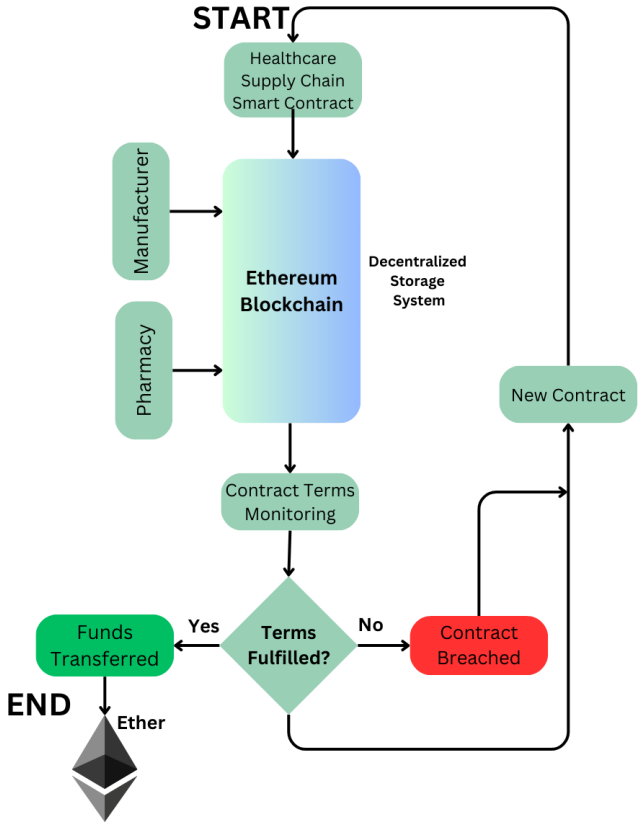


Fig. 1. Blockchain-based healthcare supply chain.

III. Implementation

The Blockchain-based healthcare supply chain is shown in Fig. 1.

A. Working of Smart Contracts

The terms of a smart contract must first be agreed upon by the participating parties. These terms are then converted into programming code, typically consisting of multiple conditional (if-then) statements. This code is stored on the blockchain network and replicated across all participating nodes. Each node executes the code, and once all network participants confirm that the predefined conditions are met, the transaction is automatically executed.

A smart contract is self-executing and includes the following basic elements:

1) Offer: The contract is initiated using Solidity with predefined conditions. The party initiating the contract provides the offer.

2) Negotiation: Upon deployment to the blockchain, the contract becomes visible to all involved parties. They may verify or negotiate the terms before finalization.

3) Approval: Once all parties agree, the trigger event (e.g., deadline, condition met) causes the contract to be signed and rendered immutable.

4) Self-satisfying conditions: Smart contracts can autonomously verify conditions using real-time data from oracles or IoT devices such as sensors and cameras.

B. Software Used

1) Solidity: A high-level programming language designed by Ethereum specifically for developing smart contracts. It compiles into machine-level code executed by the Ethereum Virtual Machine (EVM). Solidity supports variables such as boolean, integers, arrays, and modifiers, and is influenced by Python, C++, and JavaScript. It enables development of various contract types including voting, auctions, and multi-signature wallets.

2) Remix IDE: A browser-based Integrated Development Environment (IDE) for writing, compiling, and debugging Solidity smart contracts. It includes syntax highlighting, testing tools, and a built-in Solidity compiler for deploying contracts directly to the Ethereum network.

3) MetaMask: A widely used non-custodial digital wallet and browser extension that allows users to store, manage, and transact cryptocurrencies like Ether. It provides a secure, user-friendly interface for interacting with blockchain applications without requiring repeated key entries.

4) Ethereum: An open-source, decentralized platform for building and deploying smart contracts. Ether (ETH) is its native cryptocurrency, used to pay transaction (gas) fees and incentivize miners. Ethereum ensures tamper-proof, transparent, and fraud-resistant operations.

5) Ethereum Virtual Machine (EVM): The runtime environment for executing Ethereum smart contracts. EVM provides isolation between contracts, guards against denial-of-service attacks, and supports secure and traceable transactions across the decentralized network.

C. Decentralized Storage System

A decentralized storage system does not rely on a central server but instead distributes data across multiple nodes in a blockchain-enabled network. This system supports censorship resistance and immutability, allowing users to store, read, and share files in a peer-to-peer manner.

The InterPlanetary File System (IPFS) is one such decentralized storage network. It enables the publication of files, directories, and websites using cryptographic hashes stored on the blockchain. IPFS enhances security, privacy, and scalability and is often used to host web content and non-fungible tokens (NFTs).

D. Healthcare Supply Chain

The healthcare supply chain encompasses the manufacturing, distribution, and patient delivery of medical supplies and pharmaceuticals. This system must remain functional even during emergencies such as pandemics or natural disasters [17]. Blockchain enables fully digital, smart contract-based operations between healthcare providers, pharmacies, and insurers, reducing discrepancies in contracts and disputes over claims.

Key challenges include tracking inventory, expiration, shortages, and ensuring cold-chain integrity. Blockchain addresses these through immutable records that enhance transparency, prevent tampering, and authenticate pharmaceutical provenance. Each product batch can be assigned a unique identity recorded on the blockchain, allowing stakeholders to trace its journey from manufacturer to pharmacy.

Moreover, blockchain automates essential processes like order management, inventory tracking, and payments using smart contracts, reducing manual errors and enhancing efficiency. Encryption, consensus mechanisms, and access controls ensure data security and accuracy.

1) Gas cost optimization: Gas costs in smart contracts—particularly in healthcare applications—can be optimized by improving loop efficiency and minimizing unnecessary storage lookups. Direct access to storage arrays within loops significantly reduces execution costs, making smart contract deployment more feasible for large-scale healthcare systems.

E. Functions

The following are explanations of the key output functions referenced in Fig. 2 of the Healthcare Supply Chain Smart Contract:

- addDrugslot(uint256 _lot, uint256 _unitPrice): Adds a batch of drugs to the inventory for sale. Marks each medication in the lot as “For Sale” with a set price, and emits an event to notify users.
- buyDrugslot(uint256 _lot): Allows users to purchase a drug lot. Updates the buyer’s address and changes the status of each item in the lot to “Sold”, then emits a confirmation event.
- ConvertTemp(uint256 fahrenheit, uint256 celsius, uint256 _drugBatch): Converts temperatures between Fahrenheit and Celsius. Updates environmental history for each product and emits a conversion event.
- setSunSensitiveLot(uint256 _lot, bool isSunSensitive): Flags a drug lot as sunlight-sensitive using a Boolean input.
- setMedicineIngredient(string memory medicineName, string memory activeIngredient): Assigns the active ingredient to the specified medicine.
- dateformatof(uint256 _lot, uint256 _MFG_date, uint256 _EXP_date): Sets manufacturing and expiry dates for a drug lot and emits a related event.

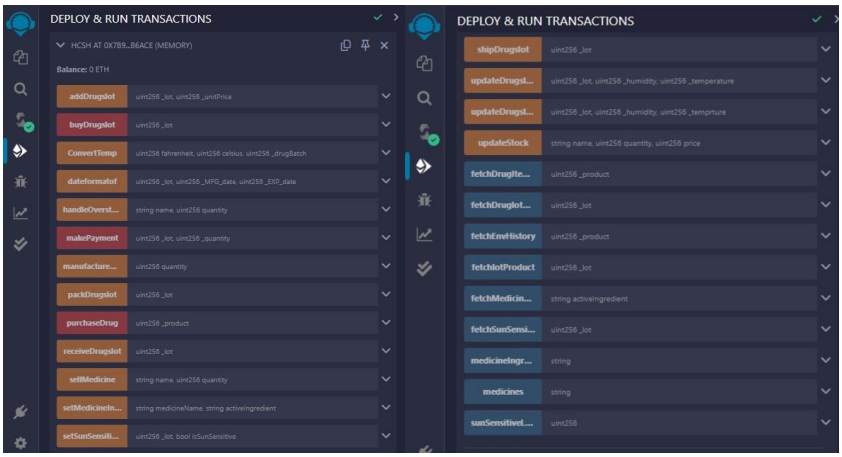


Fig. 2. Output functions.

- handleOverstock(string memory name, uint256 quantity): Manages surplus stock by flagging overstock conditions and emitting a notification event.
- makePayment(uint256 _lot, uint256 _quantity): Finalizes payment for a drug lot. Validates total price and updates status to "Purchased" upon success.
- manufactureDrugslot(uint256 quantity): Used to manufacture a drug batch. Assigns unique product IDs, updates statuses, and emits a manufacturing event.
- packDrugslot(uint256 _lot): Packages a drug lot. Confirms manufacturer identity and changes status to "Packed", recording the timestamp.
- purchaseDrug(uint256 _product): Purchases a specific drug product. Updates its status to "Purchased" and logs an event.
- receiveDrugslot(uint256 _lot): Marks a shipped lot as "Received", checks for environmental conditions, and emits a receipt confirmation.
- sellMedicine(string memory name, uint256 quantity): Sells a specified quantity of medicine. Triggers a stockout event if quantity drops to zero.
- shipDrugslot(uint256 _lot): Marks a sold lot as "Shipped" and updates the statuses of all items in the lot. Emits a shipment event.
- updateDrugslotShippmentEnv(uint256 _lot, uint256 _humidity, uint256 _temperature): Updates the environmental shipping conditions for a drug lot. Logs history and emits an event.
- updateDrugslotStockEnv(uint256 _lot, uint256 _humidity, uint256 _temperature): Modifies storage environment for a received lot. Only applicable if the drugs haven't been purchased yet.
- updateStock(string memory name, uint256 quantity, uint256 price): Updates a medicine's inventory and price in the pharmacy database.
- fetchDrugItemData(uint256 _product): View function returning product data such as ownership, manufacturer info, and price.
- fetchDruglotData(uint256 _lot): Retrieves data for a drug lot including sample item info, current owner, product count, and cost.
- fetchEnvHistory(uint256 _product): Returns environmental update history such as humidity and temperature records for a product.
- fetchlotProduct(uint256 _lot): Returns an array of product IDs associated with a given drug lot.
- fetchSunSensitive(uint256 _lot): Returns a Boolean indicating whether a lot is sunlight-sensitive.
- fetchMedicineIngredient(string memory activeIngredient): Returns the name of the medicine associated with the given active ingredient.

F. Contribution to the UN Sustainable Development Goals (SDGs)

Smart contracts contribute to achieving the United Nations Sustainable Development Goals (SDGs) in several ways:

- Goal 3 – Good Health and Well-Being: They enhance healthcare access, reduce costs, and improve quality by automating processes and reducing errors.
- Goal 9 – Industry, Innovation, and Infrastructure: Blockchain-based smart contracts promote investment in innovative healthcare systems and decentralized digital infrastructure.
- Goal 16 – Peace, Justice, and Strong Institutions: By enabling transparency and accountability, smart contracts help reduce corruption in healthcare supply chains.
- Goal 17 – Partnerships for the Goals: Through secure and verifiable data sharing, blockchain fosters stakeholder collaboration for sustainable and lasting development.

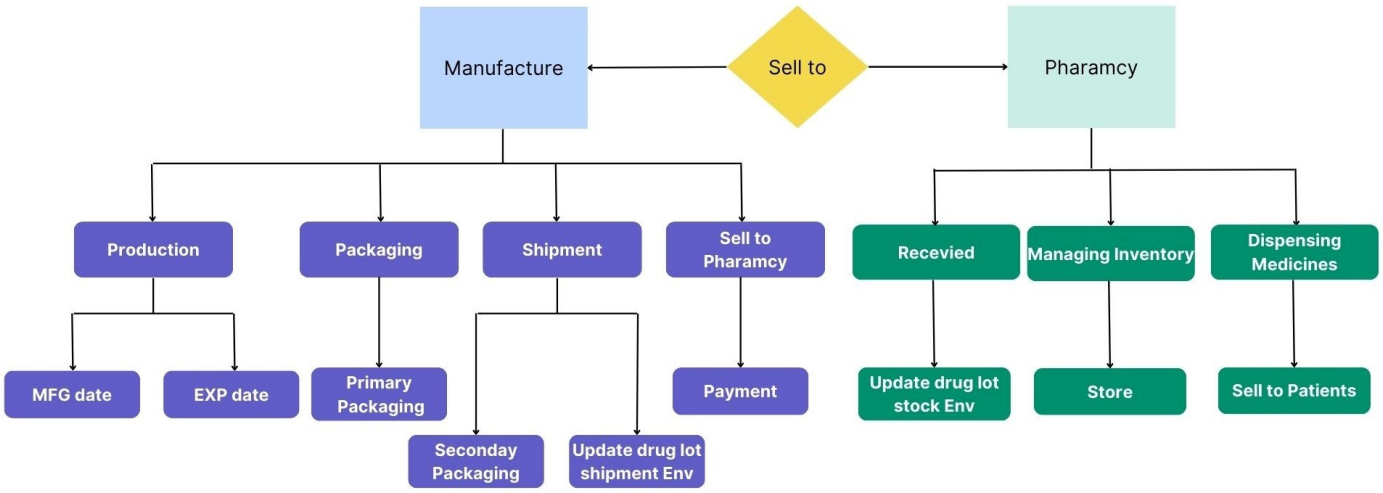


Fig. 3. The flow chart of the healthcare supply chain with all stake holders and functions.

Smart contracts and blockchain collectively offer transformative tools to support multiple SDGs across sectors, with particularly strong applications in healthcare. The flow chart of the healthcare supply chain with all stake holders and functions is shown in Fig. 3.

G. Stakeholders

Stakeholders in the healthcare supply chain include a diverse group of individuals and organizations, such as patients, physicians, nurses, pharmaceutical companies, hospitals, insurance providers, and suppliers. Each group has unique interests, requiring a well-integrated and coordinated approach.

A complex network of entities ensures the continuous flow of medical supplies and services. These collaborative networks are essential for maintaining the efficiency and quality of healthcare delivery. Efficient healthcare supply chain management reduces costs and enhances operational performance by optimizing inventory control and procurement.

Through better coordination and traceability, blockchain technology ensures high-quality patient care, prepares systems for emergencies (e.g., pandemics), and strengthens collaboration among upstream and downstream partners. It also enhances accountability, data security, and transparency in the flow of goods across the supply chain.

H. Manufacturer

Manufacturers are a vital component of the healthcare supply chain, responsible for producing and distributing medical equipment and devices. Blockchain technology offers manufacturers a powerful tool to enhance transparency, trust, and operational reliability across their business processes.

An efficient healthcare supply chain ensures the tracking and control of goods from production to wholesalers—and in some cases, directly to pharmacies or hospitals—ensuring timely and secure delivery.

1) Product traceability: Blockchain enables end-to-end product tracking, allowing manufacturers to trace items throughout the entire supply chain. This ensures product authenticity and aids in efficient recall management if necessary.

2) Supply chain efficiency: By integrating blockchain with industrial operations, manufacturers can significantly enhance process efficiency. Smart contracts can automate tasks such as payment settlements, inventory tracking, and order processing, reducing manual intervention and errors.

3) Quality control and compliance: Manufacturers can maintain immutable records of certifications, audit results, and quality control procedures on the blockchain. This not only ensures regulatory compliance but also establishes product credibility and builds customer trust.

4) Supplier management: Manufacturers rely on various suppliers for raw materials and components. Blockchain offers a secure and transparent platform for supplier verification, performance tracking, and compliance monitoring, improving supplier relationships and reducing operational risk.

I. Functions

The following are explanations of the output functions presented in Fig. 4 of the Manufacturer Smart Contract:

- assignMeAsManufacturer(): This function assigns the caller's address to the list of manufacturers using the internal function _addManufacturer. It provides the caller with the credentials required to perform manufacturer-specific tasks in the contract.
- renounceMeFromManufacturer(): This function removes the caller's address from the list of authorized manufacturers using the internal function _removeManufacturer, effectively revoking their manufacturer privileges.
- _addManufacturer(address account): An internal function that adds the specified account to the manufacturer list. Upon successful addition, it emits the

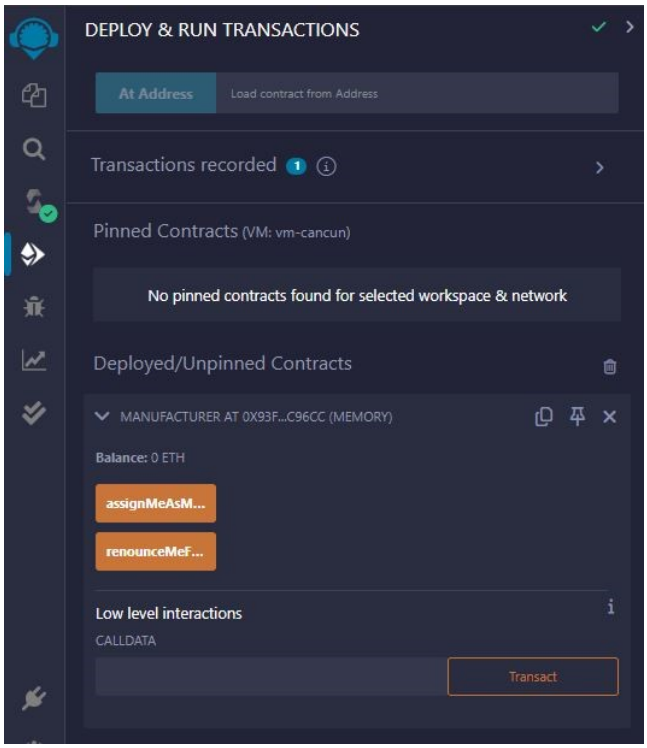


Fig. 4. Manufacturer output functions.

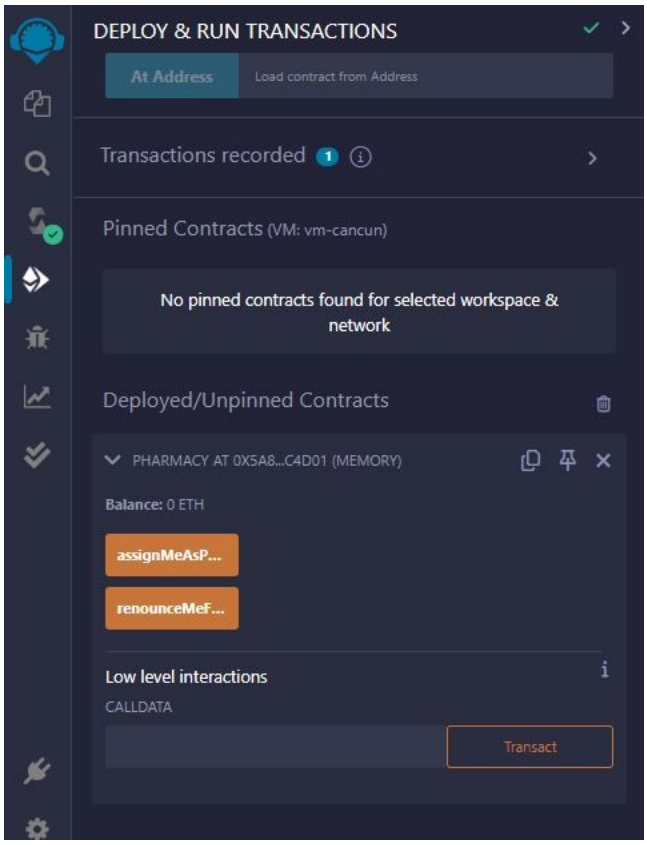


Fig. 5. Pharmacy output functions.

`ManufacturerAdded` event. This is typically used when a new manufacturer is being onboarded.

- _removeManufacturer(address account): This internal function removes the specified account from the list of manufacturers. It emits the `ManufacturerRemoved` event to confirm the action. This is typically used to revoke manufacturer access either voluntarily or administratively.

J. Pharmacy

Pharmacies form the second tier in the healthcare supply chain. They purchase pharmaceuticals and medical supplies in bulk from manufacturers. Due to high competition among pharmaceutical companies, pharmacies often have limited bargaining power and rely on wholesalers for the supply of prescription drugs to their outlets. Their functions are given in Fig. 5.

1) Medication procurement and inventory management: Pharmacies are responsible for sourcing pharmaceuticals from suppliers and maintaining sufficient inventory. Their role is crucial in avoiding shortages, minimizing stockouts, and ensuring timely access to essential medications.

2) Quality assurance and regulatory compliance: Pharmacies follow strict quality control protocols to ensure that medications remain effective and safe. This includes the handling of controlled substances, proper storage, and temperature regulation in line with regulatory standards.

3) Contribution to public health initiatives: Pharmacies play a key role in public health efforts related to drug therapy, testing, and disease screening. As accessible healthcare providers, they support disease prevention and chronic illness management at the community level.

K. Functions

The following are explanations of the output functions presented in Fig. 5 of the Pharmacy Smart Contract:

- assignMeAsPharmacy(): This public function allows any account to designate itself as a pharmacy. It uses `msg.sender` as the account reference and calls the internal function `_addPharmacy` to register the sender. This enables seamless self-assignment of the pharmacy role within the contract.
- renounceMeFromPharmacy(): This public function allows a pharmacy account to relinquish its role. It invokes the internal function `_removePharmacy` using `msg.sender` to remove the sender from the list of authorized pharmacies. It allows voluntary withdrawal from the pharmacy system.
- _addPharmacy(address account): An internal function that registers the provided address as an authorized pharmacy. It emits the `PharmacyAdded` event with the account address, and updates the contract state by granting pharmacy rights to that account.
- _removePharmacy(address account): An internal function that revokes pharmacy status for a given address. It emits the `PharmacyRemoved` event, signaling

that the specified account has been removed from the list of authorized pharmacies and no longer holds related privileges.

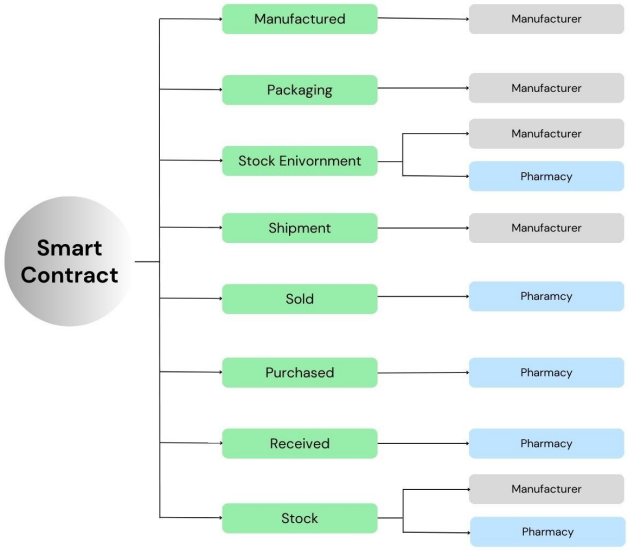


Fig. 6. System overview of blockchain based contract.

Fig. 6 shows that there are three distinct smart contracts, each having a different execution and transition cost depending on the degree of code complexity. The flowchart illustrates how registration contracts are utilized by two stakeholders: Manufacturing and Drugstore. Each stakeholder's address is recorded, and the system ensures that no duplicate addresses are used during registration. Once registration is successfully completed, the manufacturing process is initiated as the next phase.

IV. Results and Discussion

A. Cost Evaluation

Currently, there is no unified platform that enables stakeholders to exchange and communicate data at low cost. Ethereum's native cryptocurrency, ether (ETH), is used to pay for gas fees. Gas prices are expressed in gwei, which is a subunit of ETH:

- 1 ether (ETH) = 1,000,000,000 gwei (10^9 gwei)
- 1 gwei = 0.000000001 ETH (10^{-9} ETH)
- 1 ETH = 10^{18} wei (smallest unit)

The value of ETH, and thus gas cost in USD, fluctuates with market dynamics. Stakeholders must account for these costs when deploying or executing smart contracts.

B. Gas Fee

To execute any compiled smart contract on the Ethereum blockchain, a transaction fee—termed as gas—must be paid. The Ethereum Virtual Machine (EVM), responsible for processing smart contracts, consumes gas based on the computational complexity of the transaction.

Gas acts as a safeguard to prevent the network from being overloaded with overly complex or infinite-loop contracts. The more operations and storage a contract uses, the more gas it consumes.

The cost of manufacturer contract is based on the gas used, transaction cost, and execution cost, as described in Table I.

TABLE I. Cost of Manufacturer Contract

Function Name	Gas Used	Transaction Cost	Execution Cost
assignMeAsManufacturer()	25731	22374 gas	1310 gas
renounceMeFromManufacturer()	25756	22396 gas	1332 gas

Gas Cost Calculation:

$$\text{Gas Cost} = \text{Gas Used} \times \text{Gas Price}$$

The gas price depends on network congestion and user willingness to pay for faster processing.

C. Execution Cost

Execution cost is the amount of gas consumed to run the transaction or smart contract logic on the Ethereum network. This cost varies with the contract's complexity and the amount of data processed. Efficient coding practices can help reduce execution cost, which is important given that Ethereum users must pay these costs with every contract interaction.

D. Transaction Cost

The transaction cost is equivalent to the gas fee, which represents the amount of gas used to execute a transaction on the blockchain network. A transaction is required to carry out all operations, such as reading or writing data. The gas price—determined by network congestion and user willingness to pay—is multiplied by the gas consumed to calculate the total cost. Higher gas prices during peak periods can significantly affect the total cost of blockchain interaction, especially when executing complex smart contract functions.

The transaction costs for each function required to register a Pharmacy Smart Contract are displayed in Table II.

TABLE II. Cost of Pharmacy Contract

Function Name	Gas Used	Transaction Cost	Execution Cost
assignMeAsPharmacy()	25756	22396 gas	1332 gas
renounceMeFromPharmacy()	25731	22374 gas	1310 gas

Table III lists the execution costs associated with each core function in the Healthcare Supply Chain Smart Contract. These costs vary with function complexity and data manipulation.

E. Published Smart Contract Comparison

Table IV presents a gas cost comparison between our contract titled "Blockchain-enabled Smart Contracts for Logistic Control" and the published contract "Making Drug Supply Chain Secure, Traceable and Efficient: A Blockchain and Smart Contract Based Implementation" [18]. The comparison

TABLE III. COST OF HEALTHCARE SUPPLY CHAIN CONTRACT

Function Name	Gas Used	Transaction Cost	Execution Cost
addDrugslot	57879	25193 gas	2849 gas
buyDrugslot	55737	23925 gas	2721 gas
dateformatof	74768	24520 gas	3036 gas
manufactureDrugslot	203518	3983828 gas	3962624 gas
packDrugslot	58309	23907 gas	2703 gas
purchaseDruglot	32043	24010 gas	2806 gas
receiveDrugslot	36052	23907 gas	2703 gas
shipDrugslot	36090	23885 gas	2681 gas
updateDrugslotShippmentEnv	65307	24554 gas	3058 gas
updateDrugslotStockEnv	125444	24498 gas	3014 gas
handleOverstock	56275	48934 gas	27286 gas
sellMedicine	32767	25596 gas	3972 gas
updateStock	78117	67927 gas	46163 gas
setMedicineIngredient	53768	46754 gas	24826 gas
setSunSensitiveLot	53440	46469 gas	25125 gas

TABLE IV. GAS COSTS OF PUBLISHED CONTRACT IN ETH

Functionality	Our Contract	Published Contract	Published Contract (2024 Price)
Contract Deployment	0.021727111 ETH	0.002937147 ETH	0.007214388 ETH
Create Pharmacy	0.000141805 ETH	0.000443929 ETH	0.001090404 ETH
Create Manufacturer	0.000141667 ETH	0.000443862 ETH	0.001090239 ETH
Manufacture to Add Medicine	0.000555173 ETH	0.000860005 ETH	0.002112394 ETH
Pharmacy to Buy Medicine	0.000555503 ETH	0.000637359 ETH	0.001565518 ETH

includes real-time gas costs using ETH prices at two different time points: November 2022 and April 2024.

Note: The Ether price during the published contract (November 2022) was $2.2415, while in April 2024 it was $5.55057. This fluctuation directly affects cost-per-transaction, highlighting the need for gas optimization strategies in contract design.

The cost of smart contracts as well as their different functions is plotted in Fig. 7 and Fig. 8.

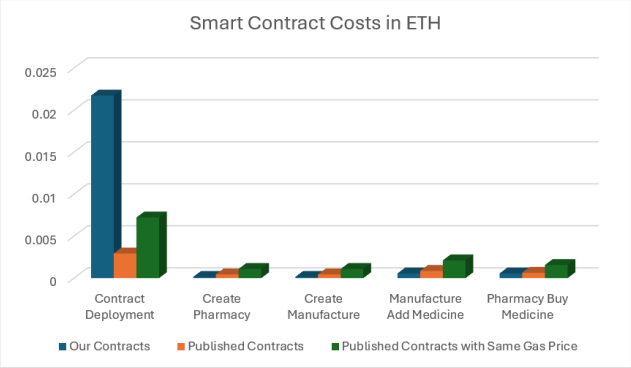


Fig. 7. Cost of smart contracts in ETH.

If we take in account of the comparison of costs in order to create multiple Manufacturers and Pharmacies, the sum of costs are illustrated in Fig. 9 and Fig. 10.

The major factors that take control of the environmental factors for the logistic control in our contract are illustrated

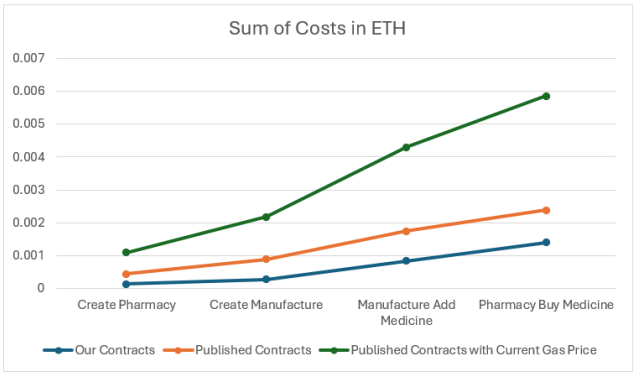


Fig. 8. Cost of different functions in smart contract.

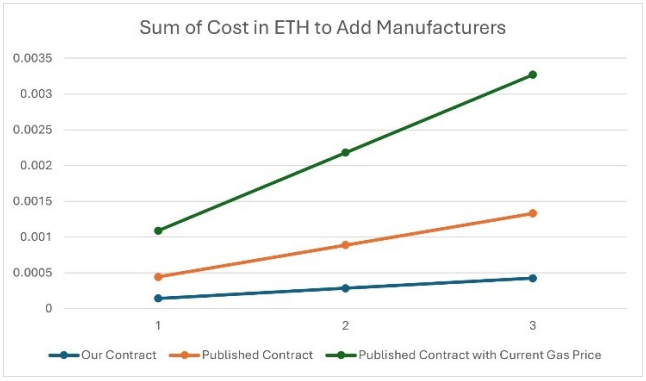


Fig. 9. Cost of adding three manufacturers in smart contract.

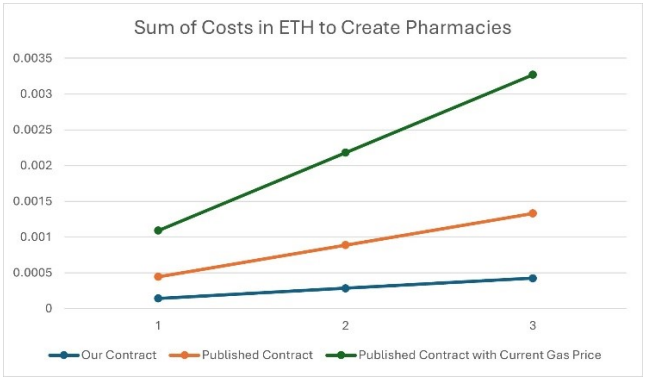


Fig. 10. Cost of adding three pharmacies in smart contract.

below with their fetching functions, by which the caller can see the data of these factors without any need of the transactions.

V. CONCLUSIONS AND FUTURE WORK

With our main goal in improving the overall gas cost and more probabilities in the environmental history of the drugs and to allow the exchange of data between multiple blockchain platforms, interoperability should become the primary area of research work since this would enable collaboration between different stakeholders. Besides the enhancement of supply chain process performance and actual time monitoring and decision-making through the electronic integration of emerging

technologies such as IoT devices with smart contracts. Like the way the Internet altered business model, blockchain has nothing but the power to overhaul value exchange and have far reaching consequences in many different fields of life. Nevertheless, the management of expectation and dealing with the difficulties will be a must-have for the businesses in the fast-changing blockchain space.

To make interoperability among blockchain platforms possible, this should be the main area of research work because this would allow partnership among the various stakeholders. It's not only the advancement of supply chain process performance and the actual time monitoring and decision-making and the electronic integration of emerging technologies such as the IoT devices with smart contracts. The way the Internet changed the business model, blockchain will have no other power to overtake the value exchange and thus very high consequences in many areas of life. However, the instances of management of expectation and dealing with the challenges will be mandatory for the businesses in the rapidly evolving blockchain world. In the context of optimizing or improving the functionality and effectiveness of the contract in terms of overall cost and more environmental factors are yet to be discovered and researched more thoroughly to implement in the supply chain contracts.

ACKNOWLEDGMENT

This work was supported by the National Research Program for Universities (NRPU), Project Number: 15363, funded by Higher Education Commission (HEC) Pakistan.

REFERENCES

[1] M. Muneeb, Z. Raza, I. U. Haq, and O. Shafiq, "Smartcon: A blockchain-based framework for smart contracts and transaction management," *IEEE Access*, vol. 10, pp. 23 687–23 699, 2022.

[2] S. Wang, L. Ouyang, Y. Yuan, X. Ni, X. Han, and F.-Y. Wang, "Blockchain-enabled smart contracts: Architecture, applications, and future trends," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 11, pp. 2266–2277, 2019.

[3] J. Sun, H.-W. Long, H. Kang, Z. Fang, A. El Saddik, and W. Cai, "A multidimensional contract design for smart contract-as-a-service," *IEEE Transactions on Computational Social Systems*, pp. 1–17, 2025.

[4] D. Mishra and S. Phansalkar, "Blockchain security in focus: A comprehensive investigation into threats, smart contract security, cross-chain bridges, and vulnerabilities detection tools and techniques," *IEEE Access*, vol. 13, pp. 60 643–60 671, 2025.

[5] G. Kumar, R. Saha, M. Conti, and W. J. Buchanan, "Leagan: A decentralized version-control framework for upgradeable smart contracts," *IEEE Transactions on Services Computing*, vol. 18, no. 3, pp. 1529–1542, 2025.

[6] X. Zhou, K. Honari, H. Liang, S. Rouhani, and S. Dick, "Blockchain-based decentralized stochastic energy management in iot-enabled smart grid with voltage regulation coordination," *IEEE Internet of Things Journal*, pp. 1–1, 2025.

[7] J. Cai, J. Chen, T. Zhang, X. Luo, X. Sun, and B. Li, "Detecting reentrancy vulnerabilities for solidity smart contracts with contract standards-based rules," *IEEE Transactions on Information Forensics and Security*, vol. 20, pp. 3662–3676, 2025.

[8] C. Xu, H. Xu, L. Zhu, X. Shen, and K. Sharif, "Enhanced smart contract vulnerability detection via graph neural networks: Achieving high accuracy and efficiency," *IEEE Transactions on Software Engineering*, vol. 51, no. 6, pp. 1854–1865, 2025.

[9] X. Zhang, Y. Song, Y. Du, C. Cai, H. Cheng, K. Xu, and Q. Li, "Smartupdater: Enabling transparent, automated, and secure maintenance of stateful smart contracts," *IEEE Transactions on Software Engineering*, vol. 51, no. 4, pp. 1266–1283, 2025.

[10] H. Huang, Z. Yin, Q. Chen, J. Zheng, X. Luo, G. Ye, X. Peng, Z. Zheng, and S. Guo, "Brokerchain: A blockchain sharding protocol by exploiting broker accounts," *IEEE Transactions on Networking*, pp. 1–16, 2025.

[11] A. R. Rajput, Q. Li, M. Taleby Ahvanooey, and I. Masood, "Eacms: Emergency access control management system for personal health record based on blockchain," *IEEE Access*, vol. 7, pp. 84 304–84 317, 2019.

[12] A. Jain, A. K. Tripathi, N. Chandra, and P. Chinnasamy, "Smart contract enabled online examination system based in blockchain network," in *2021 International Conference on Computer Communication and Informatics (ICCCI)*. IEEE, 2021, pp. 1–7.

[13] Y. Raj and S. B, "Study on supply chain management using blockchain technology," in *2021 6th International Conference on Inventive Computation Technologies (ICICT)*. IEEE, 2021, pp. 1243–1247.

[14] R. W. Ahmad, K. Salah, R. Jayaraman, I. Yaqoob, M. Omar, and S. Ellahham, "Blockchain-based forward supply chain and waste management for covid-19 medical equipment and supplies," *IEEE Access*, vol. 9, pp. 44 905–44 927, 2021.

[15] A. A. Mamun, S. Azam, and C. Gritti, "Blockchain-based electronic health records management: A comprehensive review and future research direction," *IEEE Access*, vol. 10, pp. 5768–5789, 2022.

[16] S. Yongjoh, C. So-In, P. Kompunt, P. Muneesawang, and R. I. Morien, "Development of an internet-of-healthcare system using blockchain," *IEEE Access*, vol. 9, pp. 113 017–113 031, 2021.

[17] A. Anusha, T. Shashidhar, M. Srivani, M. Srikanth, and A. Ramesh, "A blockchain-based approach for drug traceability in healthcare supply chain," *Journal of Engineering Sciences*, vol. 14, no. 4, pp. 195–209, 2023.

[18] K. C. Bandhu, R. Litoriya, P. Lowanshi, M. Jindal, L. Chouhan, and S. Jain, "Making drug supply chain secure, traceable and efficient: A blockchain and smart contract based implementation," *Multimedia Tools and Applications*, vol. 82, pp. 23 541–23 568, 2023.

A Scalable and Privacy-Preserving Hybrid Blockchain Architecture for Secure Healthcare Data Management

Sanjida Sharmin[1], Mohammad Shamsul Arefin[2], Pranab Kumar Dhar[3], Zinnia Sultana[4], Sultana Akter[5]
Department of Computer Science and Engineering,
Chittagong University of Engineering and Technology, Chattogram-4349, Bangladesh[1,2,3]
Department of Computer Science and Engineering,
International Islamic University Chittagong, Chattogram-4318, Bangladesh[1,4]
Department of Business Administration,
International Islamic University Chittagong, Chattogram-4318, Bangladesh[5]

***Abstract*—The protection of sensitive medical information has become a critical concern in modern digital healthcare. This study introduces a Hybrid Architecture that ensures secure and reliable healthcare data management through the integration of blockchain technology with off-chain and on-chain mechanisms. Patient records are encrypted using AES-256-GCM, stored in the InterPlanetary File System (IPFS), and verified using Merkle Tree structures, with only the root values anchored on Ethereum smart contracts. This design guarantees data security and integrity while achieving significant gas optimization by reducing on-chain storage costs. Experimental evaluation demonstrates that the proposed system achieves high scalability, efficient transaction processing, and strong resistance to tampering, ensuring confidentiality and auditability. By combining blockchain, cryptographic techniques, and distributed storage, the framework addresses pressing challenges of security, privacy, and trust in healthcare ecosystems. The results highlight the potential of Hybrid Architecture models to deliver a cost-effective, privacy-preserving, and scalable solution for next-generation Healthcare Data Security.**

***Keywords*—Blockchain; healthcare; AES-256-GCM; merkle tree; IPFS; data security; scalability; gas optimization; ethereum; hybrid architecture**

I. Introduction

The digitization of healthcare data has led to increased dependency on electronic health record (EHR) systems and cloud-based platforms for storage and access. However, this growing reliance introduces serious challenges related to data privacy, integrity, interoperability, and unauthorized access. Traditional centralized healthcare data management architectures are susceptible to single points of failure, cyberattacks, and restricted transparency. These limitations necessitate novel architectural paradigms that can ensure secure, verifiable, and patient-centric data handling.

Blockchain technology has emerged as a transformative solution to address these challenges by enabling decentralized and tamper-resistant systems. With its core attributes—immutability, transparency, and consensus-driven validation—blockchain can reinforce data trustworthiness and facilitate secure data sharing across distributed healthcare providers. A growing body of research has explored the integration of blockchain in healthcare, recognizing its potential to decentralize control, enhance interoperability, and safeguard sensitive clinical data [1], [2] Several works provide a foundation for understanding the structural and operational principles of blockchain. In [3], the authors presents a comprehensive overview of blockchain architecture, consensus algorithms, and anticipated evolution. Hölbl et al. [1] systematically reviewed the applicability of blockchain in healthcare, identifying both opportunities and technical limitations. Ismail et al. [4] proposed a lightweight blockchain framework tailored for healthcare environments, focusing on scalability and performance. Kakei et al. [5] explored distributed authentication infrastructures based on Hyperledger Fabric, showing potential for cross-institutional trust establishment.

Moreover, performance analysis of blockchain frameworks such as Hyperledger Fabric have shown that private blockchains can offer high throughput and low latency for medical data exchanges [6]. Khan et al. [7] discussed blockchain's synergy with the Internet of Things (IoT) and smart grid systems in healthcare. Rathi et al. [8] demonstrated the value of blockchain-enabled orchestrators for multi-domain edge computing, enabling secure data transactions at the network edge. Edited volumes such as that by Namasudra and Deka [9] further provide insights into real-world deployments, regulatory constraints, and future trends in blockchain-driven healthcare ecosystems. Despite this progress, challenges persist in effectively linking on-chain and off-chain data management. Storing complete medical files on-chain is impractical due to gas fees and blockchain size constraints. Consequently, hybrid models integrating blockchain with distributed file systems like the InterPlanetary File System (IPFS) are gaining attention [10]. These architectures allow for off-chain storage of large medical files while maintaining their integrity through on-chain hashes.

However, existing blockchain-based healthcare data management systems continue to face critical challenges, including high gas costs from on-chain storage, limited scalability when handling large datasets, and insufficient privacy-preserving mechanisms. These limitations restrict their practical adoption in real-world healthcare environments. To address these gaps, this study proposes a hybrid blockchain architecture that combines AES-256-GCM encryption, IPFS off-chain storage, and Merkle root anchoring on Ethereum, thereby ensuring strong

data confidentiality, verifiable integrity, and significant cost reduction while maintaining scalability.

This study proposes a hybrid blockchain-based framework that integrates Ethereum smart contracts and IPFS to create a secure, scalable, and tamper-resistant infrastructure for healthcare data management. Our main contributions are summarized as follows:

- A hybrid blockchain framework is proposed, where only Merkle roots of encrypted file batches are stored on-chain, achieving over 99% gas cost reduction compared to traditional methods.
- Patient records are encrypted using AES-256-GCM, ensuring data confidentiality, integrity, and secure off-chain storage.
- The system supports batch-wise Merkle tree construction and lightweight proof mechanisms, enabling scalable, high-throughput, and verifiable healthcare data management.

The remainder of this study is organized as follows: Section II reviews related work on blockchain-based healthcare data management and highlights existing challenges. Section III presents the proposed Hybrid Architecture, detailing the use of AES-256-GCM encryption, IPFS off-chain storage, Merkle Tree verification, and Ethereum smart contracts. Section IV discusses the experimental setup and performance evaluation, focusing on data security, scalability, and gas optimization. It provides a detailed discussion of the results, insights into parameter influence, and implications for practical healthcare deployment. Finally, Section V concludes the study by summarizing the contributions and outlining directions for future research.

II. Related Work

Blockchain technology has gained significant attention in healthcare for its potential to provide secure, tamper-proof data management. However, the inherent limitations in scalability pose ongoing challenges, particularly in high-volume environments, such as electronic health record (EHR) systems. Recent studies have proposed architectural enhancements and hybrid models to address these limitations.

Sadath et al. [11] introduced a hierarchical blockchain architecture tailored to healthcare, utilizing both local and global chains to minimize network congestion and improve transaction throughput. This model supports distributed data handling while preserving integrity, making it a viable solution for large-scale deployments. Similarly, Ali et al. [12] proposed a hybrid blockchain system augmented with deep learning techniques to prioritize transactions dynamically. Their model achieved improved responsiveness and reduced latency under high data loads without compromising security.

Mazlan et al. [13] provided a systematic review highlighting key scalability barriers, including throughput, latency, and interoperability. The authors discussed theoretical mitigation strategies such as Layer-2 solutions and sharding, but noted a lack of real-world implementations. In contrast, Jayabalan and Jeyanthi [14] presented a practical approach using IPFS for off-chain storage combined with on-chain metadata management, demonstrating measurable improvements in speed and scalability.

Security-focused frameworks have also shown indirect scalability benefits. Shamshad et al. [15] developed a blockchain-based EHR system with embedded access control via smart contracts. This reduced computational overhead on-chain and facilitated secure sharing. Further, Mazhar et al. [16] explored the convergence of blockchain, generative AI, and IoT in healthcare. Their study emphasized enhanced privacy, patient personalization, and interoperability, though challenges related to resource demands and decentralized storage remain.

Bathula et al. [17] conducted a comprehensive review of over 400 studies and concluded that blockchain–AI integration holds substantial promise for strengthening EHR reliability, supporting diagnostics, and managing pandemics. Collectively, the literature affirms that scalable blockchain solutions—especially those involving AI integration and off-chain data architectures—are central to building resilient, efficient, and secure healthcare infrastructures.

In the study by Taherdoost [18], the author explores the integration of blockchain technology within the healthcare sector, emphasizing its potential to enhance data privacy and security. The study provides a comprehensive overview of blockchain applications such as secure electronic health records (EHRs), patient consent management, and pharmaceutical supply chain tracking. It highlights key challenges including scalability, regulatory compliance, and interoperability with existing healthcare systems. Furthermore, the study discusses future research directions focused on improving the efficiency, adaptability, and ethical implementation of blockchain in healthcare. This work serves as a foundational reference for understanding both the opportunities and limitations of blockchain adoption in sensitive data environments like healthcare.

Kasyapa and Vanmathi [19] presents a thorough examination of blockchain integration in the healthcare domain, focusing on practical use cases, performance-related challenges, and corresponding mitigation strategies. Their investigation spans a variety of healthcare applications, including patient data management, clinical trials, and remote monitoring systems, where blockchain's decentralization and immutability offer clear benefits. The authors delve into performance bottlenecks such as latency, throughput, and energy consumption, and propose solutions involving lightweight consensus mechanisms and hybrid architectures. Their work not only showcases the transformative potential of blockchain in improving healthcare infrastructure but also provides critical insights into addressing technical and operational barriers for real-world deployment.

A comparative critique of notable blockchain-based healthcare frameworks is summarized in Table I. The table highlights the strengths and limitations of prior studies and clarifies how the proposed method advances beyond these works by integrating end-to-end encryption, scalable off-chain storage, and gas-efficient Merkle root anchoring.

III. Methodology

The proposed architecture for secure and scalable healthcare data management is structured into four sequential phases:

TABLE I. COMPARATIVE ANALYSIS OF HEALTHCARE BLOCKCHAIN FRAMEWORKS AND RELATION TO THE PROPOSED METHOD

Framework	Strengths	Weaknesses / Assumptions	Relation to Proposed Method
Sadath et al.[11]	Hierarchical chain reduces congestion; improves throughput.	Lacks end-to-end encryption; assumes scalability alone ensures privacy.	Employs AES-256-GCM encryption to ensure confidentiality while retaining scalability.
Matani et al.[20]	Batch-root commitments improve auditability and verification efficiency.	Limited scalability on large datasets; no off-chain optimization.	Uses batch-wise Merkle root anchoring with IPFS off-chain storage for scalable verification.
HealthBlock[21]	Strong decentralization and patient-centric control.	On-chain metadata storage increases gas cost and causes state bloat.	Achieves about 99.85% gas reduction by anchoring only Merkle roots on-chain.
MEDACCESSX[22]	Modular contracts improve flexibility for access control.	Complex contracts; confidentiality not guaranteed system-wide.	Maintains lightweight contract logic with strong encryption for end-to-end security.
AdaSec-Health[23]	Shard-level tuning enhances scalability.	Relies on tuning; encryption is partial or unclear.	Provides scalable hybrid storage with explicit AES-256-GCM protection.

Preprocessing, On-Chain storage, Off-Chain storage, and Verification & Decryption. Each component contributes to ensuring data confidentiality, integrity, and blockchain efficiency.The methodology involves the following key stages, as depicted in Fig. 1.

A. Dataset Preprocessing

The first stage of the proposed system involves preprocessing a structured healthcare dataset to prepare it for secure encryption, hashing, and metadata extraction. The dataset, typically stored in CSV format, contains fields such as patient_uid, PMID, file_path, title, patient, age, gender, and relevant_articles.

Table II presents representative entries from the raw dataset before preprocessing. The detailed preprocessing steps are outlined below.

1) Missing value handling: Records with null or incomplete values in critical fields (`patient_uid`, `age`, `gender`) are removed to ensure consistency and integrity throughout the encryption and verification pipeline.

2) Duplicate removal: Duplicate rows are identified and removed based on a composite key consisting of `patient_uid` and `PMID`, which uniquely represent each patient case and publication reference.

3) Age field normalization: The `age` field is often stored as a nested structure (e.g., `[[60.0, "year"]]`). A regular expression parser is used to extract the numeric value (e.g., 60) and convert it to an integer format for uniformity.

4) JSON conversion: Each row of the cleaned dataset is serialized into a standalone JSON object. These objects serve as the input to the encryption module and include metadata required for downstream file hashing and Merkle proof construction.

5) File organization: Each JSON record is saved as an individual file and assigned a filename based on its `patient_uid`. These files are stored in a staging directory for encryption and off-chain upload.

This preprocessing pipeline ensures the dataset is complete, uniquely identifiable, and structurally standardized before it enters the AES-256 encryption and Merkle tree computation stages.

B. On-Chain Processing

The on-chain component of the proposed system is responsible for storing cryptographic summaries (Merkle roots) of encrypted patient record batches to ensure tamperproof verification. The process is carried out using smart contracts deployed on the Ethereum Sepolia testnet and consists of the following steps:

1) Network connection: The system connects to the Ethereum Sepolia testnet using Infura as the remote Ethereum node provider. The Web3.py library is employed to facilitate interaction between the local Python environment and the blockchain network.

2) Smart contract integration: A pre-deployed smart contract, written in Solidity, is used to register and manage Merkle roots corresponding to batches of encrypted patient records. The contract contains methods for both storing and retrieving root values associated with batch identifiers. The smart contract behavior is formalized in Algorithm 1.

Algorithm 1 Merkle Root Commitment Smart Contract

State Variables:
  $owner$: Address of the contract deployer
  $roots$: Mapping from batch ID to Merkle root

Functions:
function CONSTRUCTOR
  $owner \leftarrow$ msg.sender
end function
function STOREMERKLEROOT(bID, $root$)
  if msg.sender $\neq owner$ **then**
    return `"Unauthorized"`
  end if
  if $roots[bID] \neq$ null **then**
    return `"Already submitted"`
  end if
  $roots[bID] \leftarrow root$
  Emit `MerkleRootStored`(bID, $root$, block.timestamp)
end function
function GETMERKLEROOT(bID)
  return $roots[bID]$
end function

3) Merkle root storage: For each batch of 1,000 encrypted files, a Merkle root is computed from their SHA-256 hashes.

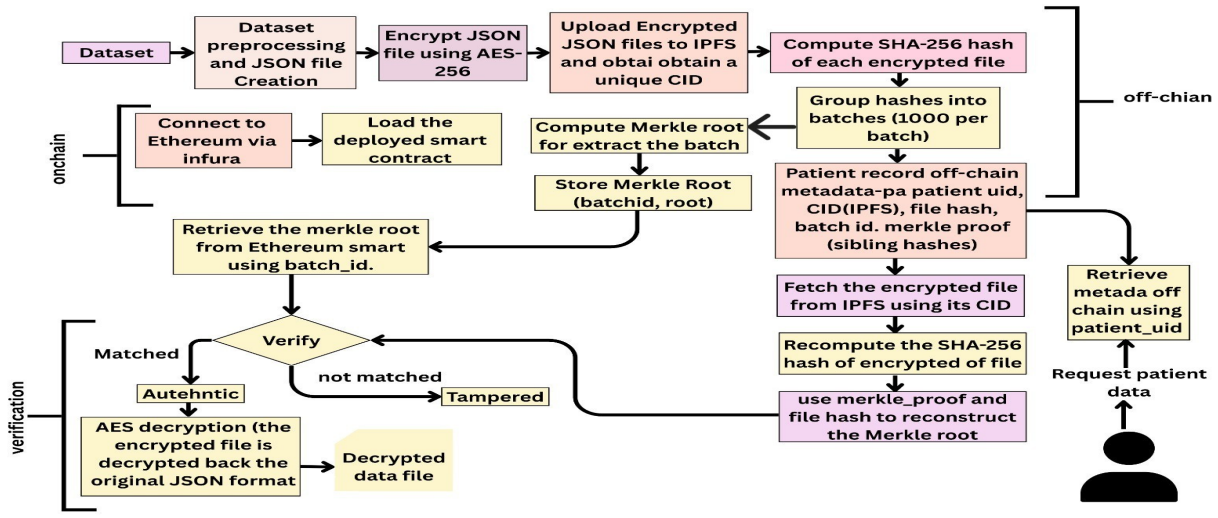


Fig. 1. System architecture of the proposed hybrid blockchain system.

TABLE II. SAMPLE INSTANCES FROM THE RAW PMC-PATIENTS DATASET

id	patient_uid	PMID	file_path	title	patient	age	gender	relevant_articles
0	7665777-1	33492400	comm/PMC...	Early Physic...	This 60-year	[[60.0, 'year']]	M	{'32320506':1, ...}
1	7665777-2	33492400	comm/PMC...	Early Physic...	A 39-year-c	[[39.0, 'year']]	M	{'32320506':1, ...}
2	7665777-3	33492400	comm/PMC...	Early Physic...	One week a	[[57.0, 'year']]	M	{'32320506':1, ...}
3	7665777-4	33492400	comm/PMC...	Early Physic...	This 69-year	[[69.0, 'year']]	M	{'32320506':1, ...}
4	7665777-5	33492400	comm/PMC...	Early Physic...	This 57-year	[[57.0, 'year']]	M	{'32320506':1, ...}

The root, along with a unique batch identifier, is then submitted to the blockchain using the smart contract function `storeMerkleRoot(batchId, root)`. This ensures immutability, auditability, and decentralized verification of the batch integrity.

C. Off-Chain Storage and IPFS Upload

The off-chain component handles encryption, decentralized file storage, hash computation, and metadata management. This process is organized into six key steps:

1) Encrypt patient JSON files: Each patient's data is serialized into JSON format and encrypted using AES-256-GCM [24], [25]. This encryption ensures both data confidentiality and integrity. The full procedure of encryption is presented in Algorithm 2.

2) Upload encrypted files to IPFS: The encrypted files are uploaded to the InterPlanetary File System (IPFS), a distributed storage network. The returned content identifier (CID) for each file is saved for future reference and verification.

3) Compute SHA-256 hash: The SHA-256 hash of each encrypted file is computed. These hashes act as the leaf nodes of the Merkle tree and serve as unique digital fingerprints of the file content.

4) Batch file hashes: File hashes are grouped into batches of 1,000 files per batch. Each batch is prepared independently for Merkle tree construction.

5) Build Merkle tree: For each batch, a Merkle tree is constructed using the SHA-256 hashes. The root of each tree is computed and serves as a compact, verifiable summary of the batch contents. The batched Merkle construction is detailed in Algorithm 3.

6) Store off-chain metadata: For each file, corresponding metadata including patient UID, CID, file hash, batch ID, and Merkle proof is compiled and stored in a structured CSV file. This metadata supports efficient file retrieval and integrity verification. To verify the integrity of any file, its SHA-256 hash is recomputed and used along with the Merkle proof to reconstruct the root hash. This reconstructed root is compared with the on-chain Merkle root associated with the batch. If the verification succeeds, the file is decrypted using the AES-256 key. This ensures that only authenticated, untampered files are accessed and utilized, thereby preserving both security and data reliability.The procedure of decryption is presented in Algorithm 4.

Algorithm 2 AES-256-GCM Encryption of Patient Records

Require: Patient record D; 256-bit AES key K; 12-byte nonce N
Ensure: Encrypted blob E containing $\{N,\ ciphertext,\ tag\}$
1: $plaintext \leftarrow \text{SERIALIZE}(D)$
2: $cipher \leftarrow \text{AESGCMINIT}(K, N)$
3: $(ciphertext, tag) \leftarrow \text{ENCRYPT}(cipher,\ plaintext)$
4: $E \leftarrow N \,\|\, ciphertext \,\|\, tag$
5: **return** E

Algorithm 3 Merkle Root Construction from Encrypted Files

Require: Encrypted files $F = \{f_1, \ldots, f_n\}$; batch size B
Ensure: Merkle roots $R = \{r_1, \ldots, r_k\}$
$H \leftarrow [\,]$ ▷ List of SHA-256 hashes
for each $f_i \in F$ **do**
  $h_i \leftarrow \textsc{Sha256}(f_i)$
  Append h_i to H
end for
Partition H into batches $\{B_j\}$ of size B; $R \leftarrow [\,]$
for each batch B_j **do**
  while $|B_j|$ is not a power of 2 **do**
    Append last(B_j) to B_j ▷ Pad with last hash
  end while
  while $|B_j| > 1$ **do**
    $L \leftarrow [\,]$
    for $k \leftarrow 1$ **to** $|B_j|$ **step** 2 **do**
      $p \leftarrow \textsc{Sha256}(B_j[k] \,\|\, B_j[k{+}1])$
      Append p to L
    end for
    $B_j \leftarrow L$
  end while
  Append $B_j[1]$ to R ▷ Merkle root for this batch
end forreturn R

Algorithm 4 AES-256-GCM Decryption of Patient Records

Require: Encrypted blob $E = N \,\|\, C \,\|\, T$; 256-bit key K; (optional) AAD A
Ensure: Decrypted patient record D
if $|E| < 28$ **then**
  return `Error: malformed input`
end if
$N \leftarrow E[0:12]$ ▷ 12-byte nonce
$T \leftarrow E[-16:]$ ▷ 16-byte tag
$C \leftarrow E[12:-16]$ ▷ ciphertext
$cipher \leftarrow \textsc{AesGcmInit}(K, N)$
if A provided **then**
  $\textsc{AesGcmUpdateAAD}(cipher, A)$
end if
$plaintext \leftarrow \textsc{DecryptAndVerify}(cipher, C, T)$
if verification fails **then**
  return `Error: authentication failed`
end if
$D \leftarrow \textsc{ParseJson}(plaintext)$
return D

IV. Results and Evaluation

This section presents the experimental setup, evaluation metrics, and performance analysis of the proposed hybrid blockchain architecture. The system is evaluated for gas efficiency, latency, throughput, scalability, and data security using real-world datasets.

A. Experimental Setup

The proposed framework was implemented using Python 3.x and deployed in Google Colab for off-chain processing. SHA-256 hashing and AES-256-GCM encryption were performed using the `hashlib` and `cryptography` libraries, respectively. IPFS Desktop and `go-ipfs` were used for decentralized file storage, while Web3.py and Infura were employed to interact with smart contracts on the Ethereum Sepolia testnet. Smart contracts were developed using Solidity and deployed via the Hardhat framework.

Experiments were conducted on a computing system equipped with an Intel(R) Core(TM) i7-4510U CPU operating at 2.00 GHz (with a turbo frequency of 2.60 GHz) and 6 GB of RAM, running on Windows 10 Pro (64-bit). The proposed hybrid blockchain architecture was implemented and evaluated using the Ethereum Sepolia testnet, accessed via the Infura gateway. Encryption of patient records was performed using the AES-256-GCM algorithm to ensure both data confidentiality and integrity. The experimental dataset consisted of 39,936 encrypted patient records in JSON format, which were processed and managed across the off-chain and on-chain components of the system.

B. Evaluation Metrics

The performance of the proposed system is evaluated using five core metrics: gas cost, latency, throughput, scalability, and data security. Two of these metrics—latency and gas consumption—are further analyzed mathematically to validate the framework's efficiency.

1) Latency modeling: The total system latency (L_{total}) is the cumulative delay incurred during data preprocessing, storage, and verification phases. The overall system latency (L_{total}) can be modeled as the sum of initiation, confirmation, and off-chain delays, as formulated in Eq. (1):

$$\begin{aligned} L_{\text{total}} = {} & (t_{\text{initiation}} + t_{\text{confirmation}}) \\ & + (t_{\text{IPFS_upload}} + t_{\text{IPFS_retrieval}}) \\ & + (t_{\text{hash_compute}} + t_{\text{hash_compare}}) \end{aligned} \tag{1}$$

where, $t_{\text{initiation}}$ and $t_{\text{confirmation}}$ represent the time taken for transaction submission and blockchain block confirmation, respectively. $t_{\text{IPFS_upload}}$ and $t_{\text{IPFS_retrieval}}$ correspond to the delays associated with decentralized storage operations in IPFS. Finally, $t_{\text{hash_compute}}$ and $t_{\text{hash_compare}}$ denote the time required for computing file hashes and verifying file integrity.

Empirical results show that blockchain transaction latency ($L_{\text{on-chain}}$) averages around 14.3 seconds, while IPFS upload and retrieval ($L_{\text{off-chain}}$) contribute an additional 2 to 4 seconds. As shown in Eq. (2), the overall system latency is modeled as the sum of on-chain and off-chain delays.

$$L_{\text{total}} \approx L_{\text{on-chain}} + L_{\text{off-chain}} \tag{2}$$

Thus, the total latency for the hybrid blockchain–IPFS framework is approximately in the range of 16 to 18 seconds, validating the framework's suitability for non-real-time healthcare applications.

2) Throughput modeling: Throughput (T_{system}) is defined as the rate at which encrypted patient records are processed, verified, and committed within the system. It is a critical performance metric that reflects the system's efficiency in handling large-scale healthcare datasets. As expressed in Eq. (3), the system throughput is defined as the ratio of the total number of records to the overall system latency.

$$T_{\text{system}} = \frac{N_{\text{records}}}{L_{\text{total}}} \tag{3}$$

where, N_{records} is the number of encrypted records in a given batch and L_{total} is the total latency (as defined in Equation 1) required to process and verify the batch.

3) Gas cost matrices: In blockchain-based systems, particularly those utilizing Ethereum smart contracts, the transaction execution cost is quantified through gas consumption. As shown in Eq. (4), the gas cost in Ethereum is determined by multiplying the gas used by the gas price.

$$\text{Gas Cost (ETH)} = \text{Gas Used} \times \text{Gas Price (ETH)} \quad (4)$$

where, Gas Used denotes the actual number of gas units consumed during transaction execution, and Gas Price (ETH) is the market price of one unit of gas, typically denominated in Gwei (where $1\,\text{Gwei} = 10^{-9}\,ETH$).As formulated in Eq. (5), the gas cost in USD is obtained by multiplying the gas cost in ETH with the current ETH price in USD.

$$\begin{aligned}\text{Gas Cost (USD)} = (&\text{Gas Used} \times \text{Gas Price (ETH)}) \\ &\times \text{ETH Price (USD)}\end{aligned} \quad (5)$$

C. Gas Cost Efficiency

In Ethereum-based smart contract systems, the cost of executing a transaction is determined by gas consumption. The proposed method significantly reduces gas costs by batching encrypted file hashes into Merkle trees and storing only the Merkle root on-chain. As defined in Eq. (6), the total gas consumption is obtained by multiplying the number of batches with the gas required per root.

$$G_{\text{total}} = N_{\text{batches}} \times G_{\text{per_root}} \quad (6)$$

where,

- $N_{\text{batches}} = 40$ is the number of file batches.
- $G_{\text{per_root}} = 30,000$ gas is the average gas consumed to store a single Merkle root on-chain.

Substituting into Eq. (6), the total gas cost becomes Eq. (7):

$$G_{\text{total}} = 40 \times 30{,}000 = 1{,}200{,}000 \text{ gas} \quad (7)$$

TAs shown in Eq. (8), the total gas consumption for processing all files individually is computed as:

$$\begin{aligned}G_{\text{individual}} &= N_{\text{files}} \times G_{\text{per_hash}} \\ &= 39,936 \times 20,000 \\ &= 798,720,000 \,\text{gas}\end{aligned} \quad (8)$$

Using this value, the gas savings ratio is then derived in Eq. (9) as:

$$\eta_{\text{gas}} = 1 - \frac{G_{\text{total}}}{G_{\text{individual}}} = 1 - \frac{1,200,000}{798,720,000} \approx 0.9985 \quad (9)$$

This demonstrates that the proposed system achieves a remarkable gas savings ratio of approximately 0.9985, which corresponds to a reduction of nearly 99.85% in gas cost compared to executing all transactions, individually.

D. Gas Cost Analysis in Ethereum

The following parameters:

- Total gas used to store 40 Merkle roots: $G_{\text{total}} = 1{,}200{,}000$ gas
- Gas price: 30 Gwei = 30×10^{-9} ETH
- ETH/USD exchange rate: \$3,000

The total ETH cost is computed as:

$$\begin{aligned}\text{Cost}_{\text{ETH}} &= 1{,}200{,}000 \times 30 \times 10^{-9} \\ &= 0.036\,\text{ETH}\end{aligned} \quad (10)$$

Then, converting to USD:

$$\text{Cost}_{\text{USD}} = 0.036 \times 3000 = \$108 \quad (11)$$

The total ETH cost is computed, as shown in Eq. (10), and then converted into USD using Eq. (11).

E. Latency Analysis

Fig. 2 illustrates the latency distribution across 40 batches of encrypted patient records during off-chain processing and on-chain Merkle root submission. Each batch consists of approximately 1,000 encrypted files. The measured latency includes file encryption, IPFS upload, SHA-256 hash computation, Merkle tree construction, and Ethereum transaction confirmation.

The latency across batches ranges from approximately 11.4 seconds to 28.1 seconds, with most batches falling within the 18 to 26 second interval. Variations are primarily attributed to network fluctuation during IPFS storage and Ethereum block confirmation delays. Notably, the final batch exhibits the lowest latency due to a reduced number of files.

This consistent and bounded latency behavior demonstrates the system's ability to maintain predictable performance while processing large-scale healthcare datasets, making it suitable for near real-time and secure health data workflows.

F. Throughput Performance

Throughput is defined as the number of encrypted patient records processed per second, including encryption, IPFS upload, Merkle tree construction, and Merkle root submission. Fig. 3 illustrates the throughput achieved across 40 batches of encrypted files, each consisting of approximately 1,000 records.

As shown, throughput ranges from approximately 36 to 74 files per second, depending on batch size, system load, and network conditions. Batches with fewer files (e.g., Batch 40) naturally exhibit higher throughput due to reduced processing

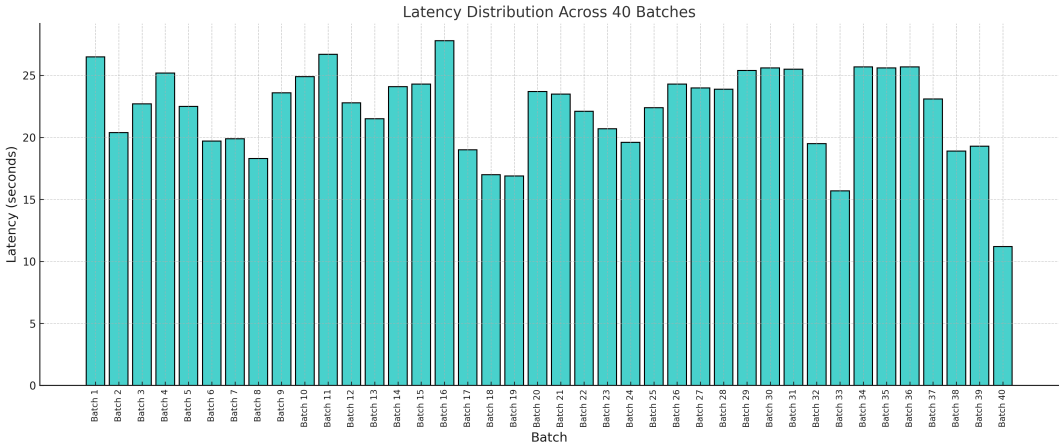


Fig. 2. Latency distribution for 40 batches of encrypted patient records.

time. The consistent throughput across most batches reflects the stability of the proposed system under typical workloads and validates its ability to handle large volumes of healthcare data efficiently.

G. Off-Chain Scalability Analysis

To evaluate the system's scalability in terms of storage distribution, the off-chain encrypted files were grouped into 40 batches of approximately 1,000 files each. Fig. 4 illustrates the estimated storage size per batch in megabytes (MB).

As shown, most batches maintain a consistent size of approximately 194 to 195 MB, demonstrating balanced storage allocation and predictable performance. The final batch, containing fewer files, exhibits a slightly lower size, validating the linear scaling behavior of the proposed system. This confirms that batch-based off-chain file management is both scalable and storage-efficient for large-scale healthcare datasets.

H. Security Assurance

AES-256-GCM encryption provided confidentiality and integrity through its built-in authentication tag. In conjunction with Merkle proof-based integrity verification, the system offers a secure mechanism for handling sensitive healthcare data in decentralized environments.

I. Comparative Analysis with Prior Works

Table III presents a comparative analysis of the proposed hybrid blockchain system against three recent healthcare blockchain frameworks: HealthBlock [21], Sadath et al. [11], and Matani et al. [20]. The comparison focuses on ten key criteria, including storage architecture, on-chain data strategy, encryption, scalability, and verification methodology.

Unlike existing models that either store full metadata on-chain or rely on non-scalable storage mechanisms, the proposed system leverages off-chain IPFS storage and stores only 40 Merkle roots on-chain—one for each batch of 1,000 encrypted patient records. This significantly reduces on-chain data bloat while preserving verifiability through Merkle proofs. The use of AES-256 encryption further ensures end-to-end confidentiality and integrity, which is either partially implemented or entirely missing in the compared systems.

In terms of scalability, the proposed framework supports high-volume batch uploads and demonstrates high data scalability. It adopts a public Ethereum testnet (Sepolia), offering better decentralization and accessibility compared to permissioned or hybrid networks like Fabric and sharded blockchains. Furthermore, it incorporates a robust security model combining AES encryption, Merkle proof, and access control lists (ACLs), making it well-suited for large-scale, privacy-sensitive healthcare environments.

J. Comparative Results Across Datasets

The differences observed in comparative results across datasets can be attributed to the architectural limitations of existing frameworks. Approaches such as Sadath et al.[11] and Matani et al.[20] perform adequately on smaller datasets but exhibit reduced scalability at higher volumes due to limited batching and lack of off-chain storage. HealthBlock emphasizes decentralization but incurs significant gas costs from on-chain metadata storage, which becomes inefficient as data size grows. In contrast, the proposed method consistently maintains scalability and efficiency across datasets by integrating AES-256-GCM encryption, IPFS off-chain storage, and batch-wise Merkle root anchoring. These design choices make the algorithms particularly well-suited for structured and semi-structured healthcare data, including electronic health records, demographic information, prescriptions, and transaction logs. For unstructured or multimedia data such as medical imaging or continuous IoT sensor streams, additional preprocessing into metadata or compact representations is necessary, as direct storage may increase retrieval latency.

1) Latency comparison with existing methods: Fig. 5 presents a comparative latency analysis between the proposed method and three existing blockchain-based healthcare frameworks: Matani et al. [20], Sadath et al. [11], and HealthBlock [21]. The x-axis represents the number of transactions, while the y-axis shows latency in seconds.

The proposed method consistently demonstrates the lowest latency across all transaction scales (1,000 to 6,000), ranging

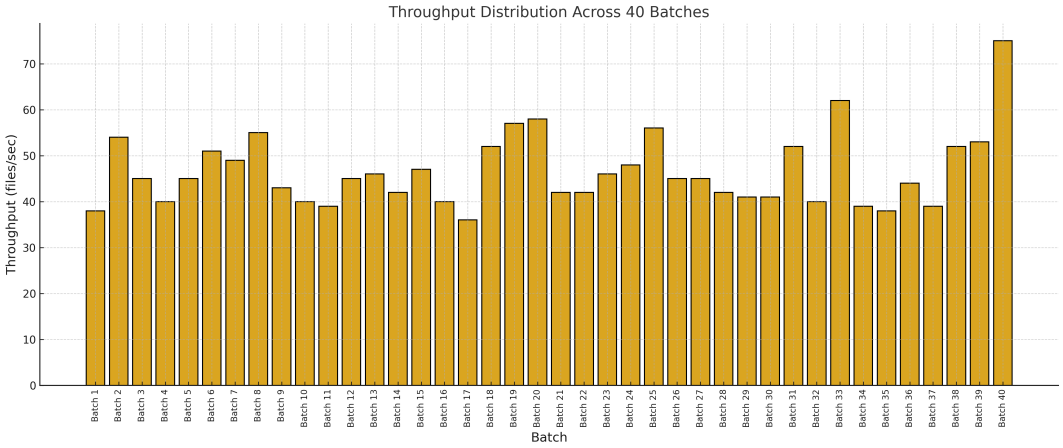


Fig. 3. Throughput distribution across 40 batches, expressed in files per second.

TABLE III. COMPARATIVE ANALYSIS OF THE PROPOSED HYBRID BLOCKCHAIN SYSTEM AGAINST PRIOR WORKS

Criteria	Proposed System	HealthBlock [21]	Sadath et al. [11]	Matani et al. [20]
Storage	IPFS Remote (∼180 MB/batch)	OrbitDB + IPFS	CouchDB	Shard-distributed
On-Chain Data	40 Merkle Roots	Per-file metadata	Batch root	Per-level TX hashes
Hashing	SHA-256	SHA-256	SHA-256	SHA-256
Merkle Tree	Yes (40 roots)	No	No	Used in scoring phase
Encryption	AES-256	Partial (OrbitDB/IPFS)	No encryption	No encryption
Batch Uploads	1,000 per batch	No	Yes	Implied via sharding
Blockchain Type	Public (Ethereum Sepolia)	Private (Fabric)	Private (Fabric)	Hybrid Sharded
Verification Type	Merkle proof	Log traceability	Proof of computation	Hash-linked consensus
Data Scalability	High	Low	Medium	Medium
Security Model	AES + Merkle + ACL	N/A	Metadata only	Hierarchical + scoring

TABLE IV. COMPARATIVE ANALYSIS OF GAS COST AND ARCHITECTURE: PROPOSED SYSTEM AGAINST PRIOR WORK

Criteria	Proposed System	MEDACCESSX [22]	AdaSec-Health [23]
Blockchain Type	Public (Ethereum Sepolia)	Private Ethereum Testnet	Public (Ethereum EIP-1559)
Storage Strategy	Off-chain IPFS + Merkle Root anchoring	On-chain smart contracts for access + data metadata	IPFS + ECOA-optimized shard blockchain
On-Chain Data	40 Merkle Roots for 39,936 files	User, access, data, and policy contracts (UMC, ACC, etc.)	Dynamic block/meta parameters via ECOA
Gas Optimization Method	Merkle batching, minimal on-chain state	Modular contracts but fixed logic	Shard tuning: validation degree, block size, interval
Total Gas Used	1.2M gas for 39,936 files	3.1M gas for deployment (all contracts)	Avg. marginal cost = 0.47 gas units per transaction
Gas Cost in USD	$108 @ 30 Gwei, $3K/ETH	$154.65 (same rate)	Not directly priced, implied savings
Gas Savings (%)	99.85% vs. per-file storage	Not explicitly reported	Reported via performance gains under ECOA
Smart Contract Complexity	Minimal (root commit/retrieve only)	High (RBAC, ABAC logic, events)	Medium to high (governance, consensus, voting)
Batch Handling	1,000 per batch	Per access/role	Per 4-block shard optimization
Security Model	AES-256-GCM + Merkle + ACL	Access logs on-chain + ABAC/RBAC	ECOA-based scoring + CP-ABE + reputation
Scalability Focus	Gas-efficient IPFS and Merkle design	Functional control granularity	Adaptive shard configuration for forks/delay

from 1.8 to 3.2 seconds. This efficiency is attributed to the use of Merkle root batching, AES-256-GCM encryption, and lightweight on-chain commitment. In contrast, the methods by Matani et al.[20] and Sadath et al.[11] exhibit significantly higher and more variable latency, primarily due to per-transaction or per-batch on-chain data handling without

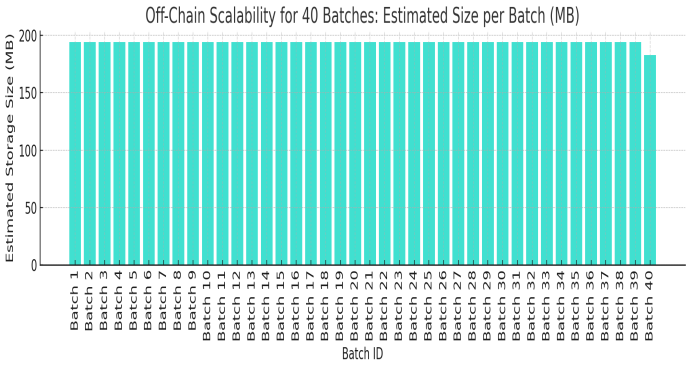


Fig. 4. Off-chain scalability analysis: estimated file size distribution across 40 batches.

optimized batching. HealthBlock performs moderately but lacks Merkle-based verification and full encryption, leading to comparatively higher overhead.

This result confirms that the proposed system is well-suited for secure healthcare applications where low-latency processing is essential for real-time or near-real-time interactions.

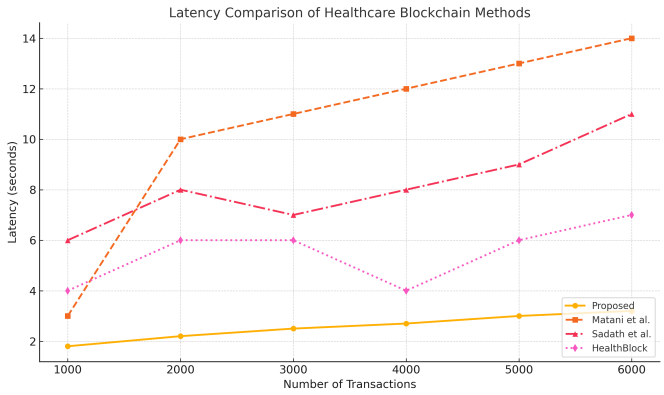


Fig. 5. Latency comparison of the proposed hybrid blockchain system with existing methods across increasing transaction volumes.

2) Throughput comparison with prior methods: Fig. 6 illustrates the throughput performance of the proposed method compared to existing blockchain-based healthcare systems developed by Matani et al. [20], Sadath et al. [11], and HealthBlock [21]. Throughput is measured as the number of transactions successfully processed per second across increasing transaction volumes.

The proposed method demonstrates superior scalability, with throughput values rising linearly from approximately 560 to over 1,850 transactions per second, as the number of transactions increases from 1,000 to 6,000. This is attributed to the efficient batch processing design, lightweight Merkle root anchoring, and minimal on-chain data overhead. In contrast, existing approaches such as Matani et al. and Sadath et al. show relatively static or non-linear throughput behavior due to more extensive on-chain storage and lack of optimization in data handling.

HealthBlock performs moderately well but exhibits inconsistent throughput beyond 4,000 transactions. The consistent and scalable performance of the proposed system confirms its suitability for high-throughput healthcare data environments.

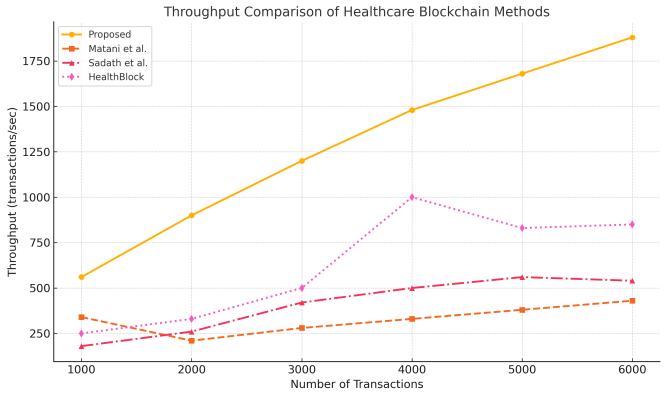


Fig. 6. Throughput comparison of the proposed system with existing methods under varying transaction volumes.

3) Gas cost comparison with prior healthcare blockchain frameworks: Table IV provides a detailed comparative analysis of the proposed hybrid blockchain system against two notable healthcare blockchain frameworks—MEDACCESSX and AdaSec-Health. The evaluation focuses on gas optimization techniques, contract architecture, batch processing strategy, and scalability models.

K. Influence of Parameters

The influence of algorithmic parameters has been systematically examined to highlight their role in the performance of the proposed architecture. The batch size has a direct impact on gas optimization: larger batches significantly reduce transaction costs by minimizing the number of on-chain commitments, although they may introduce a slight increase in latency during Merkle root computation. The AES-256-GCM encryption parameters ensure robust data security while imposing only negligible overhead on processing time, making them suitable for large-scale deployments. In contrast, IPFS upload and retrieval times contribute substantially to off-chain latency, as formalized in Eq. (1) and Eq. (2). The gas price in the Ethereum network strongly influences the overall deployment cost; however, the batching mechanism employed in this study demonstrates resilience against network congestion and fluctuating gas conditions. Finally, the Merkle tree depth, inherently determined by batch size, affects proof generation and verification complexity. Collectively, these observations underscore the necessity of careful parameter tuning to achieve a balanced trade-off among scalability, security, and efficiency in real-world healthcare data management scenarios.

V. Conclusion

The findings demonstrate that the proposed hybrid blockchain framework achieves a practical balance between privacy preservation, scalability, and cost efficiency in healthcare data management. By integrating AES-256-GCM encryption, IPFS-based off-chain storage, and Merkle root anchoring on Ethereum, the system ensures end-to-end confidentiality and

verifiable integrity while reducing gas consumption by nearly 99.85% compared to conventional on-chain storage.

An important observation is that off-chain operations—particularly IPFS upload and retrieval—dominate overall system latency. This suggests that optimizing distributed storage through improved node configurations, caching strategies, or integration with Layer-2 storage solutions may yield greater performance benefits than adjusting blockchain parameters alone.

In comparison with existing healthcare blockchain frameworks, the proposed design introduces novelty by simultaneously ensuring confidentiality, integrity, and scalability in a cost-effective manner. This positions the framework as a practical and viable solution for strengthening trust, security, and reliability in digital healthcare ecosystems.

Future work will extend this study by deploying the framework in real-world blockchain environments, integrating decentralized identity mechanisms for stronger patient-centric control, and implementing adaptive access control for secure multi-stakeholder sharing. Additional directions include exploring Layer-2 scalability enhancements, supporting unstructured healthcare data such as medical imaging and IoT streams, and enabling cross-chain interoperability to facilitate secure record exchange across healthcare institutions.

REFERENCES

[1] M. Hölbl, M. Kompara, A. Kamišalić, and L. N. Zlatolas, "A systematic review of the use of blockchain in healthcare," *Symmetry*, vol. 10, no. 10, p. 470, 2018.

[2] S.-K. Kim and H. C. Vong, "Secured network architectures based on blockchain technologies: A systematic review," *ACM Computing Surveys*, 2025.

[3] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *2017 IEEE international congress on big data (BigData congress)*. Ieee, 2017, pp. 557–564.

[4] L. Ismail, H. Materwala, and S. Zeadally, "Lightweight blockchain for healthcare," *IEEE access*, vol. 7, pp. 149 935–149 951, 2019.

[5] S. Kakei, Y. Shiraishi, M. Mohri, T. Nakamura, M. Hashimoto, and S. Saito, "Cross-certification towards distributed authentication infrastructure: A case of hyperledger fabric," *IEEE Access*, vol. 8, pp. 135 742–135 757, 2020.

[6] G. Al-Sumaidaee, R. Alkhudary, Z. Zilic, and A. Swidan, "Performance analysis of a private blockchain network built on hyperledger fabric for healthcare," *Information Processing & Management*, vol. 60, no. 2, p. 103160, 2023.

[7] F. A. Khan, M. Asif, A. Ahmad, M. Alharbi, and H. Aljuaid, "Blockchain technology, improvement suggestions, security challenges on smart grid and its application in healthcare for sustainable development," *Sustainable Cities and Society*, vol. 55, p. 102018, 2020.

[8] V. K. Rathi, V. Chaudhary, N. K. Rajput, B. Ahuja, A. K. Jaiswal, D. Gupta, M. Elhoseny, and M. Hammoudeh, "A blockchain-enabled multi domain edge computing orchestrator," *IEEE Internet of Things Magazine*, vol. 3, no. 2, pp. 30–36, 2020.

[9] S. Namasudra and G. C. Deka, *Applications of blockchain in healthcare*. Springer, 2021, vol. 83.

[10] S. Sharmin, I. H. Sarker, M. Shamim Kaiser, and M. S. Arefin, "Interplanetary file system-based decentralized and secured electronic health record system using lightweight algorithm," in *Proceedings of the International Conference on Big Data, IoT, and Machine Learning: BIM 2021*. Springer, 2021, pp. 691–702.

[11] L. Sadath, D. Mehrotra, and A. Kumar, "Scalability performance analysis of blockchain using hierarchical model in healthcare," *Blockchain in Healthcare Today*, vol. 7, pp. 10–30 953, 2024.

[12] A. Ali, H. Ali, A. Saeed, A. Ahmed Khan, T. T. Tin, M. Assam, Y. Y. Ghadi, and H. G. Mohamed, "Blockchain-powered healthcare systems: enhancing scalability and security with hybrid deep learning," *Sensors*, vol. 23, no. 18, p. 7740, 2023.

[13] A. A. Mazlan, S. M. Daud, S. M. Sam, H. Abas, S. Z. A. Rasid, and M. F. Yusof, "Scalability challenges in healthcare blockchain system—a systematic review," *IEEE access*, vol. 8, pp. 23 663–23 673, 2020.

[14] J. Jayabalan and N. Jeyanthi, "Scalable blockchain model using off-chain ipfs storage for healthcare data security and privacy," *Journal of Parallel and distributed computing*, vol. 164, pp. 152–167, 2022.

[15] S. Shamshad, K. Mahmood, S. Kumari, C.-M. Chen *et al.*, "A secure blockchain-based e-health records storage and sharing scheme," *Journal of Information Security and Applications*, vol. 55, p. 102590, 2020.

[16] T. Mazhar, S. khan, T. Shahzad, M. A. khan, M. M. Saeed, J. B. Awotunde, and H. Hamam, "Generative ai, iot, and blockchain in healthcare: application, issues, and solutions," *Discover Internet of Things*, vol. 5, no. 1, p. 5, 2025.

[17] A. Bathula, S. K. Gupta, S. Merugu, L. Saba, N. N. Khanna, J. R. Laird, S. S. Sanagala, R. Singh, D. Garg, M. M. Fouda *et al.*, "Blockchain, artificial intelligence, and healthcare: the tripod of future—a narrative review," *Artificial Intelligence Review*, vol. 57, no. 9, p. 238, 2024.

[18] H. Taherdoost, "Privacy and security of blockchain in healthcare: Applications, challenges, and future perspectives," *Sci*, vol. 5, no. 4, p. 41, 2023. [Online]. Available: https://www.mdpi.com/2413-4155/5/4/41

[19] M. S. B. Kasyapa and C. Vanmathi, "Blockchain integration in healthcare: A comprehensive investigation of use cases, performance issues, and mitigation strategies," *Frontiers in Digital Health*, vol. 6, p. 1359858, 2024. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fdgth.2024.1359858/full

[20] A. Matani, A. Sahafi, and A. Broumandnia, "Improving scalability in blockchain systems using multi-level sharding based on heterogeneity of network nodes," *Computing*, vol. 107, no. 2, pp. 1–34, 2025.

[21] B. Zaabar, O. Cheikhrouhou, F. Jamil, M. Ammi, and M. Abid, "Healthblock: A secure blockchain-based healthcare data management system," *Computer Networks*, vol. 200, p. 108500, 2021.

[22] G. Shi, M. Qi, Q. Zhong, N. Li, W. Gao, L. Zhang, and L. Gao, "Medaccessx: A blockchain-enabled dynamic access control framework for iomt networks," *Sensors*, vol. 25, no. 6, p. 1857, 2025. [Online]. Available: https://doi.org/10.3390/s25061857

[23] R. Alghamdi, S. Qamar, F. K. Shaikh, and M. A. Alqarni, "Improving data transmission through optimizing blockchain sharding in cloud–iot based healthcare applications," *Egyptian Informatics Journal*, vol. 26, no. 2, pp. 123–135, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1110866525000544

[24] K. P. N. Rao and S. Chinnaiyan, "Blockchain-powered patient-centric access control with midc aes-256 encryption for enhanced healthcare data security," *Acta Informatica Pragensia*, vol. 13, no. 3, pp. 374–394, 2024. [Online]. Available: https://aip.vse.cz/pdfs/aip/2024/03/03.pdf

[25] Y. Wang *et al.*, "Aes-256 encryption for secure file storage in blockchain systems," *International Journal of Innovative Research in Technology*, vol. 11, no. 10, pp. 4488–4490, 2020. [Online]. Available: https://ijirt.org/publishedpaper/IJIRT174494_PAPER.pdf

Advanced Strategies for Big Data Resource and Storage Optimization: An AI Perspective

Ayoub Sghir, Ayoub Allali, Najat Rafalia, Jaafar Abouchabaka
Department of Computer Science, Faculty of Sciences, Ibn Tofail University, Kenitra, Morocco

***Abstract*—The increasing use of advanced technologies with artificial intelligence in our daily lives has become an urgent necessity to facilitate tasks in a fluid and simple way, which leads to the generation of huge amounts of data. This data comes from various sources: media, social networks, connected objects., online transactions, and smart devices, among other sources. These data are generally organized into three categories: structured, unstructured, and semi-structured. This data is therefore called Big Data. This data is characterized by its enormous size and fast flow, as well as by the diversity of its sources. The importance of data lies in its ability to provide future perspectives and improve the decision-making process. To get the most out of this data, it must be stored and processed, but current technologies face many challenges and are often insufficient to cope with the huge amounts of data generated. It is necessary to look for advanced and highly efficient technologies, capable of storing the entirety of the data and processing it faster. We can also rely on artificial intelligence to help improve the use of storage and processing resources by compressing data or deleting excess data, thus saving storage space. This study discusses various approaches for optimizing Big Data processing, such as the use of AI compression techniques, the PSNR-SSIM method, and many others. The compression ratio for these algorithms is around 90%. With these technologies, it is possible to optimize the use of storage space, ensuring efficient and optimized management.**

***Keywords*—Artificial intelligence; big data; optimization; resource; storage**

I. Introduction

Big data has become the most widely used term in our digital world, particularly with the advent of the fifth generation and the use and reliance on machine learning and the Internet of Things. The Big Data industry is growing so fast that, according to a report by IDC (International Data Corporation), by 2025, around 30% of all data will be generated in real-time. According to the same entity, with around 150 billion connected devices worldwide in 2025, most of them will generate data in real-time, according to IDC. The global data domain is expected to grow from 23 zettabytes (ZB) in 2017 to 175 zettabytes (ZB) in 2025.

Today, numerous studies highlight the need to develop advanced and robust storage systems to effectively manage Big Data on a large scale [1]. Scalability considerations include reliability and availability as key technology design objectives for Big Data storage [2]. For this reason, these technologies have been designed to handle huge amounts of data, with their diverse sources and diversities, at high processing speeds.

Storage systems include Hadoop, NOSQL, Cloud Computing, NEWSQL and Data Lake. What sets these technologies apart is their ability to scale by adding more nodes to the cluster, as well as their ability to use a distributed file system across multiple machines for Hadoop. NoSQL databases also offer great flexibility for various data models (key-value, document, column, graph), enabling non-relational data to be managed with high performance. Cloud computing is also distinguished by its ability to adapt to demand, increasing storage capacity as needed, while offering rapid access to this massive data. Each technology has the capacity to store and process huge quantities of data.

Despite the power of these technologies, they also pose significant challenges, particularly with the growing sophistication of data flows. These challenges include the rapid growth of Big Data, the high cost of storage, real-time processing, data retention policies, and data duplication.

These challenges include the rapid growth of Big Data, the high storage cost, real-time processing, data retention policies and data duplication. All these challenges have led to the need to look for innovative solutions that can store and process all this data in real-time without consuming significant resources. Among these solutions are emerging technologies such as quantum storage, DNA data storage and holographic storage, a major revolution in the data storage industry, but at present, they are not the ideal solution, especially as most of them are still in the development phase. experimental framework and will not be accessible to everyone, as they will have high costs and complex management.

The best solution remains to work on developing and improving current technologies by enhancing the storage capacity of certain technologies to accelerate the pace of fast and immediate storage and processing, as well as integrating artificial intelligence into storage management frameworks for resource-constrained systems. This enables efficient storage utilization and improves system performance. AI-based solutions can be applied to improve resource utilization and system performance, in particular, their ability to learn and make decisions. The main objective of this study is to present techniques for optimizing Big Data.

This study proposes four techniques for optimizing storage, two of which are based on artificial intelligence (multi-agent systems and K-means clustering), batch processing, and other approaches based on statistical indicators. The objective of this project is to examine the effectiveness of these techniques on real data stored in MongoDB databases, focusing on four key aspects of the storage process: compression time, final size, access time, and CPU and RAM usage. Our findings highlight the strengths and weaknesses of each method and offer specific guidance for optimizing the efficiency, performance, and sustainability of storage in Big Data systems.

This study is organized as follows: In Section II, we

provide an overview of the concept of Big Data along with the available technologies for its management and storage. Section III introduces some of the problems faced by data, especially at the storage level. Section IV describes our simulation methodology and the environment in which it operates. Section V details the role of AI for optimization. In Section VI, we present and discuss our simulation results. Finally, Section VII concludes this work and outlines future work possibilities.

II. Related Works

To improve dynamic storage management [7] this paper outlines different approaches for improving storage management to facilitate communication between AI agents in decentralized systems. It provides a multilayer communication environment that allows AI agents to dynamically identify communication needs to improve communication speed and efficiency. Decentralized storage systems are also compared in the paper and their performance is evaluated with regard to the efficiency of data storage, retrieval, and transfer between AI agents. This study aims to improve the scalability and interoperability of AI systems in decentralized environments by focusing on reducing communication delays.

In this paper [11], the authors present a model for improving data storage in smart grids, using an improved simulated annealing algorithm. This model aims to solve the problems of longitudinal and horizontal penetration between multi-level data centers in the information transmission networks of smart grids. A comparison of mathematical algorithms is also presented in the paper, based on the context of optimizing data storage in the Smart Grid. The aim is to evaluate the effectiveness of these algorithms in managing resources in a distributed computing system and to determine which algorithm offers superior performance in terms of task processing speed and efficient resource allocation.

In article [17], a method is examined for improving distributed storage using multi-agent systems. Its aim is to tackle the challenges of distributed storage, such as resource optimization and data management in a fragmented environment. To solve these problems, this method relies on multi-agent systems in which each agent manages a part of the overall task. The aim is to optimize storage efficiency while reducing costs and access times, minimizing obstacles, and improving overall system performance.

This article [21] presents a secure and efficient method for storing and managing data, using an intelligent optimization algorithm (IOA-DSAM). Images are divided into several parts, then encrypted using homomorphic encryption (HE). The improved fruit fly optimization algorithm (MFFO) improves encryption key generation. Once the shares have been received, they are decrypted and reconstructed. In this article, the author presents the various stages of the suggested model, such as share generation, the image encryption model and the more efficient key generation process. It also presents experimental results that compare the performance of the IOA-DSAM algorithm with other models. The results of the article offering better performance, thus proving its effectiveness for securing and managing images in a digital environment. The article concludes that this technique improves data security while offering optimized storage management.

In [25], a deep learning-based framework for storage management was proposed. This solution offers the ability to dynamically adjust key system parameters, such as cache size and queue depth. This method leads to performance improvements in terms of transfer rate, latency, and CPU utilization. However, this method focuses on optimizing system configurations without considering other crucial aspects such as compression, size, reads, and CPU/RAM. Furthermore, it does not address comparisons with other technologies and solutions. In the same context, in this article [13], particularly with regard to storage technologies, the author proposes a multi-agent approach to classify data as “hot” and “cold” in an HDFS cluster. He uses the classification to adapt redundancy and compression, with the aim of optimizing resource utilization. However, this method is confined to a specific Hadoop framework and does not systematically evaluate performance. Most research focuses on a single criterion, such as access time, system configuration, or a specific method, without adopting a comprehensive multi-criteria approach, leaving a gap in the literature on storage optimization. Furthermore, there is little research that systematically compares AI methods with conventional techniques in a practical application setting. Furthermore, validation often focuses on specific environments or technologies (such as HD file systems or system caches), which limits the ability to generalize. Our contribution offers a fair comparative study of four techniques, judged on four main axes, contributing to a more balanced and useful analysis of storage optimization.

III. Challenges in Big Data Storage

Data storage issues - The high cost of data storage and retention: Many companies have invested billions of dollars in data center development to build out the infrastructure elements of Big Data, due to the high cost of data storage and retention. In recent years, Amazon has invested $35 billion in AWS data centers in one region alone [14]. In the context of the Internet of Things, or with the fifth generation, the accumulation of information emitted by sensors generated by devices can quickly lead to data processing problems, especially as the available storage space is limited. This can lead to the loss of data or the deletion of readings, restricting data storage and monitoring capacity, and making data analysis extremely complex.

Data management and processing: The set of operations involved in metadata processing includes the collection, storage, cleansing, and management of huge quantities of data, whether in different forms or from different sources. Data ingestion into a data lake or streaming processing engine is one of the steps involved in Big Data processing. Although advanced data collection and processing mechanisms are available, there are certain obstacles, such as the lack of sufficiently advanced techniques or tools to extract and process data efficiently in real-time or close to real-time. There is still a lack of fast, efficient processing in current solutions. there is also a concern about data integration, as it is difficult to merge data from different sources and formats. Particularly given the complexity of processing algorithms, which require constant adjustments and improvements, especially when it comes to variations in formats and structures, resulting in prolonged integration times. It is also worth mentioning the importance of

data consistency. A major problem during processing. Errors or inconsistencies can occur when updating data.

Data privacy and security: Numerous studies have explored the existence of numerous personal information breaches and data security vulnerabilities in Big Data environments. Hence the urgent need for real-time security measures. The study [16] highlights the complex nature of data security in the Big Data landscape while recognizing the need for innovative solutions that go beyond traditional security models. Delving deeper into data protection, it becomes clear that data security in Big Data is a multifaceted challenge that requires a deep understanding of the unique attributes of large datasets. As organizations continue to harness the power of Big Data, data control is a major challenge in addressing the security issues inherent in this transformative era [15]. Data protection is essential in the broader context of information management because information is sensitive. Rigorous and vigilant security measures are essential to prevent unauthorized access, data breaches, and misuse. This involves the application of specific information laws, which encourage the judicious use of data without touching or divulging personal or confidential information. Strong security systems and protocols are essential.

IV. Optimization Technique

Optimization is the process of making something, such as a system or process, more effective and efficient. By implementing strategies and techniques to optimize storage, we can reduce storage costs and improve data processing capabilities. Big Data storage management refers to the process of storing, structuring, and managing large quantities of data to facilitate retrieval and analysis. There are several techniques for managing and improving Big Data storage [9]. As illustrated in Fig. 1 below, here are the key methods for optimizing data storage in the context of Big Data:

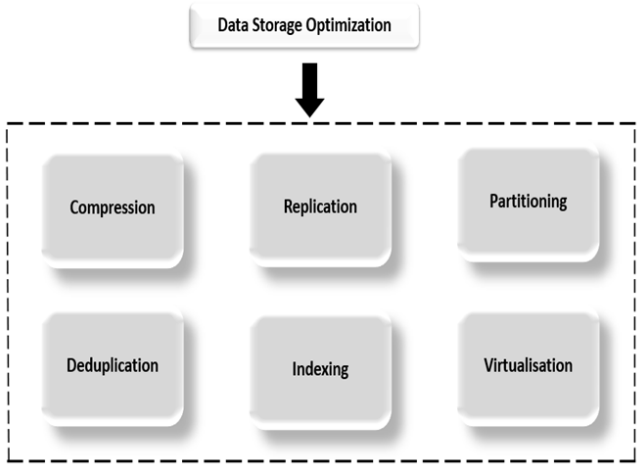


Fig. 1. Data storage optimization.

1) Compression: The compression process, among the most effective methods of optimizing data storage, is the use of coding to avoid data storage and transmission, which facilitates data transmission [18], restructuring, and other modifications to reduce the size of digital data files without altering their essential characteristics. The main aim of compression is to reduce the number of bits needed to code the data while reducing the memory required to store the data [19]. Compression can be divided into two categories: lossless and lossy. Lossless compression preserves the original data. While lossy compression reduces the original data size by compromising certain details. It permanently eliminates unnecessary bits. Although this only removes irrelevant information, it may result in the loss of some information due to compression. Lossy compression is often used for multimedia files such as audio, images, graphics, and video.

2) Replication: In a distributed environment, replication is frequently used to store multiple copies of the same data in different locations on multiple servers. When data is needed, it is retrieved from the nearest copy, thus avoiding delays and improving system performance [20]. Data replication offers the advantage of accelerating data access, reducing access times, and improving data availability. Replication is frequently used in distributed databases, cloud storage systems, and networks to improve consumer response speed and ensure service continuity.

3) Partitioning: A partitioning model is one of the most common methods used to optimize storage, and consists mainly of dividing a large dataset into smaller parts for easier handling, making it easier to manage large quantities of data and control their arrangement on disks. Each partition contains a part of the data, and these parts are usually spread over several servers or systems. These partitions are generally scattered over different database tables and contain their subset of data. It is possible to consider each partition as an independent database, but they are always part of the same logical database. Data partitioning can be achieved in several different ways. These include horizontal and vertical partitioning. The former is often used for the table mechanism. All partitions contain a separate part of the rows of a set of information. In a database, for example, records can be divided into partitions according to a specific range of values. In terms of vertical division, these partitions group together various subsets of columns. This is the opposite of horizontal division. All other types of partitioning are used to improve performance, whether for operations, aggregations or fast filters. In addition, they enable data to be distributed across several servers, enabling parallel processing and improving performance and utilization of computing resources.

4) Deduplication: Deduplication eliminates redundant data [22], to reduce the amount of data stored in a Big Data environment. Storage and storage optimization remain major challenges, especially with the exponential growth of data. After checking the data on the storage system, deduplication is used to increase and optimize storage space. As redundant data is written to storage, the deduplication process begins to be identified and eliminated. Duplicate data is identified, deleted, and replaced by a pointer to the first iteration of the data block. Deduplication also used the block verification technique, after dividing the data into blocks. A comparison is made between each block and other blocks already stored in the storage technologies to detect duplicates. With this method, users can separate specific workloads and quickly retrieve the updated backup, thus improving data storage efficiency.

5) Virtualization: Optimization through virtualization involves the creation of virtual structures for Big Data systems,

such as servers, storage, or networks. Data virtualization can offer platforms that exploit all the information they collect to achieve different objectives. Data virtualization is a data management method that enables an application to retrieve and manipulate information without the need for technical details about the data, such as its source format or physical location [23]. About storage, it offers the possibility of adding more physical resources to a more flexible storage space, facilitating the dynamic localization of data. It improves efficiency by optimizing the use of resources and space. Virtualization makes storage and retrieval tasks simpler, further optimizing the storage infrastructure.

V. Role of AI for Optimization

While confronting the challenges of storage management methods in resource-constrained systems, there is an urgent need to look for more dynamic and fluid solutions. One such solution is artificial intelligence, which can store data intelligently, classify and label it, and retrieve it quickly and easily. The study establishes the fundamental concepts of Big Data and machine learning, highlighting their interrelationships. It then examines the various ways in which Big Data improves machine learning algorithms, such as by improving data diversity, which promotes better generalization and reduces overlearning. In addition, Big Data uses artificial intelligence to obtain more precise information.

Artificial intelligence (AI) is a technology based on large databases and multiple algorithms, enabling the rapid discovery of new cultural systems and the creation of cognitive maps through data collection, data mining, and resource integration [6]. Simply put, AI aims to simulate human cognitive functions. AI represents a broad branch of computer science that seeks to create intelligent machines capable of performing tasks that usually require human intelligence [5]. Artificial intelligence plays a crucial role in storage and resource optimization. It enables machine-learning algorithms to reduce data duplication, remove redundancy, compress data, and load-balance to ensure optimal use of storage resources and dynamically compress information [7]. Analyzing historical data makes it possible to predict resource allocation and forecast future requirements. Artificial intelligence manages distributed storage environments, ensuring optimum use of resources. Finally, artificial intelligence simplifies data collection and resource allocation according to priorities and access patterns.

There are four technologies, illustrated in Fig. 2, which we will examine in this section.

1) Machine learning: is one of those technologies that has had a lot of success. To keep up with the expansion of the machine learning sector, some of the things that influence it, such as infrastructure and technical capabilities, also need to be increased [4]. Machine Learning is a branch of AI that enables machines to learn without the need for programming. Machine Learning is associated with Big Data, as it deals with data. Machine learning algorithms can be trained on datasets to spot patterns, relationships, and trends. In the field of storage management, machine learning algorithms can be trained on historical access patterns to predict future data requirements, identify the most efficient compression techniques for particular data types, and adjust real-time caching and retrieval strategies to suit system conditions [8].

2) Deep Learning (DL): is a branch of artificial intelligence that draws on the workings of the nervous system of living beings to solve more complex problems [24]. Deep learning algorithms process different types of information in a similar way to the way our neural networks react to the neural signals assigned to them. Deep learning is one of the recently recognized methods for storing and intelligently managing huge amounts of data. Thanks to deep learning, it operates on sophisticated models. Information can be reduced by identifying and eliminating repetitions, while preserving its reliability and quality.

In addition to optimizing the use of storage resources using deep learning algorithms, it is possible to optimize input/output (I/O) processing, thereby reducing access and disk access times. In addition, the use of neural networks to identify patterns and repetitions in data enables high-performance compression without compromising quality, thus optimizing the storage space used [25]. Deep learning reduces costs and optimizes the performance of Big Data management systems. This is particularly beneficial in a situation where the volume of data produced and collected continues to grow, requiring sophisticated, intelligent storage solutions.

3) Reinforcement learning: is an artificial intelligence technique that reproduces the human learning process to achieve its objectives. The learning paradigm used in RL is based on the ability of agents to make decisions by interacting with a context. The rewards or penalties the agent receives enable it to adapt its actions over time. This information-based method can be used to make dynamic storage choices in response to the changing requirements of Big Data systems. Several aspects of RL focus on optimizing Big Data management, including adaptive data positioning and real-time traffic management [3]. RL offers the possibility of dynamically adapting the choice of storage nodes based on available capacities and network conditions. This method is able to balance the load between servers and increase network throughput by adopting adaptive choices for each stage. RL is also used to improve the dynamic positioning of information in various storage strata, based on access patterns and the specific characteristics of the data, such as its temperature and input frequency, among other criteria [10]. The aim of this method is to optimize system performance by reducing processing time and resource expenditure.

4) A Multi-Agent System (MAS): is a distributed structure made up of various autonomous, intelligent software entities, referred to as agents. Agents interact with each other by transmitting information, and then collaborate with other agents to complete complex tasks. The decentralized approach of multi-agent systems (MAS) optimizes Big Data storage. Each agent, possessing learning skills, is able to make independent decisions regarding the management and processing of information. Coordination enables actors to optimize the use of storage resources by harmonizing workloads and dynamically modifying management policies in response to disparities between demand and resource availability.

As a key result, multi-agents facilitate the optimal distribution of data across multiple storage points, increase access speed and reduce throughput thanks to information compression and deduplication methods. Artificial intelligence enables these players to set up a decentralized structure and

distributed storage [7-25] to exchange information efficiently with each other. This ensures immediate access to information. Agents are also able to adjust their procedures based on client and system performance, with the aim of selecting the most sophisticated technology for their underlying storage devices and ensure astute information management with the SMA monitor, which enables Data files to be accessed and classified as hot or cold [17]. Agents also have the ability to optimize storage system operation by optimizing replication and reacting to changes in data usage. This will reduce floor space and optimize resource utilization, while maintaining the right balance between efficiency and protection.

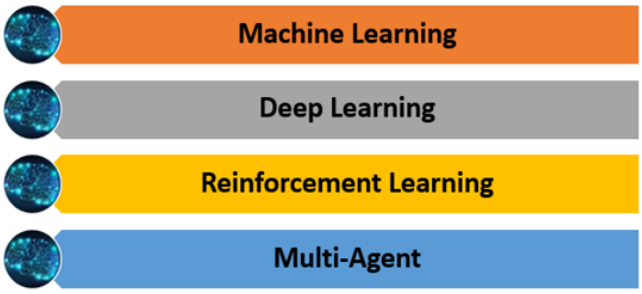


Fig. 2. AI-Driven techniques for big data storage optimization.

VI. Study of Methods and Results

There are four main stages in the Big Data management process, one of which is storage. One of the main challenges in Big Data is storage. They perform efficient, fluid storage of various forms of data while ensuring that fast access and real-time processing performance are maintained. It is essential to have an infrastructure and systems capable of storing and retrieving data rapidly without requiring considerable storage space. In this section, we look at the different optimization models used to improve Big Data storage, which achieved the best results based on four criteria (compressed size, compression time, read time, and average resources).

The data was chosen from the group of plant images, with a size of 900 MB. This is a specific unstructured data, which is considered as a type of Big Data. NoSQL databases were selected to store the data. MongoDB, a technology that manages data in the form of documents, offers a perfect solution for storing image-type data. It provides a distributed file system specially designed for MongoDB, called GridFS, specifically designed to store and retrieve image files. The storage optimization process is divided into three stages, as shown in Fig. 3:

When selecting the optimization algorithm, four algorithms are available. Two of these are based on the intelligent approach, namely multi-agent and K-means. The other algorithm is based on compression using batch processing and the PSNR-SSIM method. Phases 2 and 3 involve implementing the principles of the algorithms to improve storage in MongoDB technology.

The first method, entitled Smart Optimization and Batch Processing, is an algorithm that uses hybrid JPEG compression first, followed by gzip and batch processing. Its aim is to

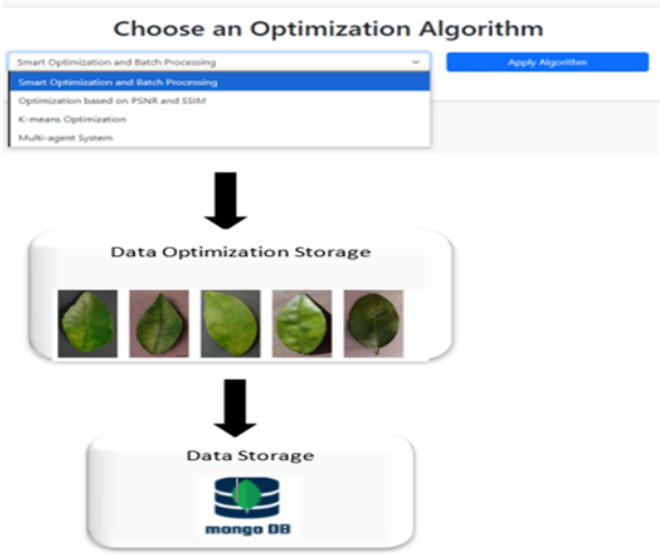


Fig. 3. Process architecture storage optimization.

divide images into several groups rather than processing them individually or all at once, thus effectively reducing image size and managing large numbers of images without overloading memory.

The second algorithms also use compression, but this time the PSNR-SSIM metric is used. As we know, pixels are small squares that represent the image. Digital image analysis is the specialty of uniform-density image processing. The image pixel matrix is two-dimensional. The numerical values of these pixels vary between 0 and 255. Color images are encoded on 3 RGB channels (red, green, and blue), with an integer ranging from 0 to 255 for each channel. To assess the quality of an image after compression, the PSNR-SSIM principle combines two metrics: the first PSNR measure evaluates the signal-to-noise ratio. It is measured in decibels (dB). The quality of the compressed image is improved as the PSNR value increases, while SSIM evaluates the structural conformity between the original image and the compressed image. Often, an SSIM greater than 0.85 is perceived as satisfactory quality for visual images.

Each time images are compressed; the process starts by calculating the PSNR and SSIM values between the original and compressed images, as soon as PSNR exceeds 30 dB and SSIM exceeds 0.85. The process is complete when all 70,000 data images have been processed. There are also two intelligence-based methods: unsupervised K-means and multi-agent.

The third K-means method is based on minimizing the number of colors present in the image and allows datasets to be grouped into K distinct clusters, based on a color grouping based on the specified number of clusters. Each group corresponds to a group of similar images. The K-means algorithm converts images into a single dimension and then groups them into K clusters based on the specified number of clusters. Each cluster corresponds to a group of similar images, and the distance between each image and the center of its cluster is minimized. Clustering aims to group similar images

together for optimal processing and compression. Once the images have been grouped, each group is processed separately.

The multi-agent approach is the fourth method. This method implements autonomous agents that work together to perform tasks related to data compression and optimization management in a big data context. We employed three agents. Each agent has his or her own responsibilities. The first agent is responsible for applying random compression levels to images that are run on a regular basis. When activated, this agent compresses an image and sends the compressed file to the second agent. In this way, compressed files are received and saved in MongoDB GridFS. The role of the third agent is to supervise and monitor system performance. All these autonomous agents are defined to perform specific tasks in a coordinated way at different compression levels to reduce image size.

The four suggested algorithms were grouped and presented as diagrams. During the optimization process, the performance of the algorithms was evaluated in terms of size, compression time, read time, RAM usage, and CPU consumption.

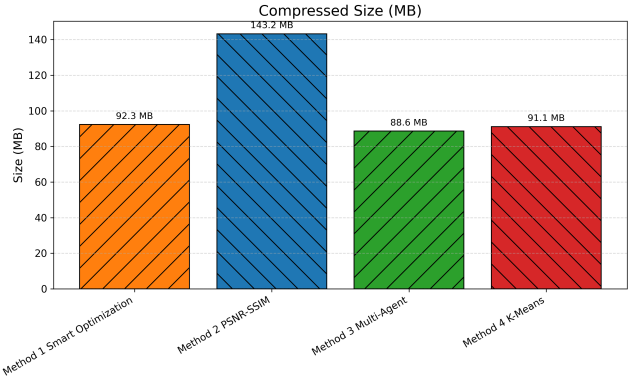


Fig. 4. Compressed size for all 4 methods.

Following the optimization of the four techniques, the results are illustrated in Fig. 4. This figure summarizes the graphs corresponding to the four techniques: Each instance produced a file size significantly smaller than that of the initial file. Technique 2 produced a 140 MB document, with a compression ratio of approximately 84.44%. This indicates that compression reduced the data size by 84.44%. The other methods, SMA and K-means, performed best (88.6 MB and 91.6 MB), achieving a compression rate of approximately 91%. In contrast, PSNR-SSIM produced a larger compression size (143.2 MB), thus reducing its effectiveness in contexts where storage space is critical. The results of the four methods indicate highly effective compression, with a higher compression index. Thus, our multi-criteria study indicates that examining compressed size alone is not sufficient to judge a method; it must be combined with other parameters such as the time required for compression, the time required for reading, and resource utilization in order to make a rigorous choice in the context of Big Data.

The time required to compress the data was considered as a second criterion, and the image data was correctly compressed and stored by the storage system. However, according to Fig. 5, the two artificial intelligence algorithms achieved a

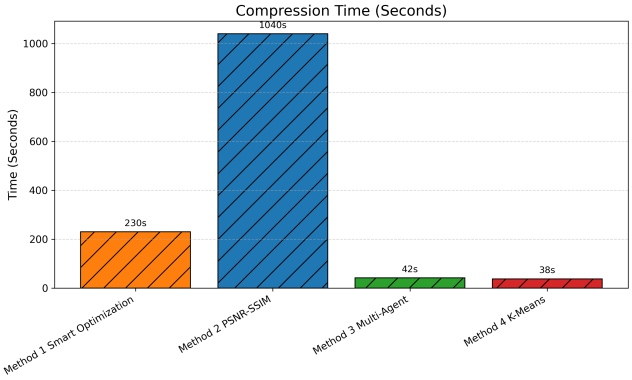


Fig. 5. Compression time for all 4 methods.

compression time of 31 and 26 seconds, respectively, for multi-agent method 3 and K-means method 4. It is observed that the AI-based methods require little time to compress the data, while the other methods require more time. Methods 3 and 4 demonstrate considerably higher processing efficiency.

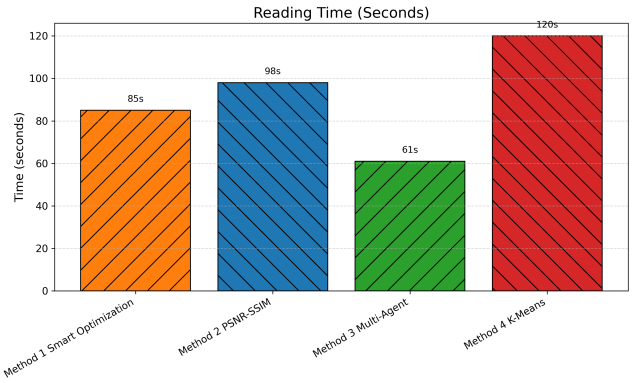


Fig. 6. Reading time for all methods.

The third criterion evaluated in this study was data reading time (see Fig. 6). For all four methods, reading time ranked well against traditional methods. However, there is one method that scored well for data reading, namely the multi-agent method. This method required just 60 seconds to read compressed data from MongoDB (GridFS file management system). This compares favorably with other techniques, which are more time-consuming to varying degrees.

The execution time analysis reveals that the SMA technique is fast, making it a suitable option for contexts, where regular access to data is essential. The Smart-Optimization technique, which takes 85 seconds, offers a satisfactory balance between performance and flexibility. However, although the K-means method offers remarkable compression performance, it suffers from a major drawback in terms of read time, which is 120 seconds. The PSNR-SSIM technique, with an average time of 98 seconds, is average but less efficient than the SMA method.

In Fig. 7, resource performance was evaluated and two comparisons were made for the four methods. The illustration compares CPU and RAM usage for four optimization techniques. Intelligent optimization and PSNR-SSIM require

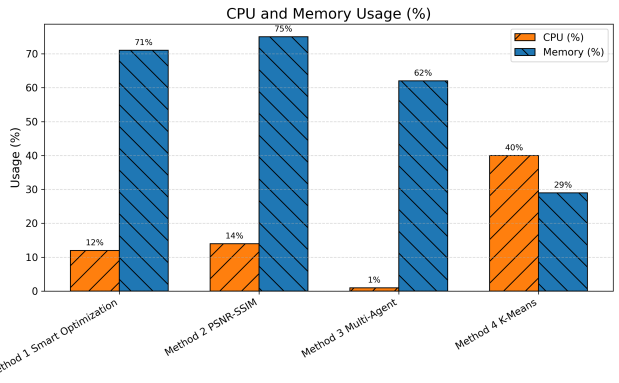


Fig. 7. CPU and RAM usage for all methods.

a lot of memory (70%), but little CPU (14%). Multi-Agent uses only 0.4% of the CPU while utilizing 62% of the RAM, demonstrating optimal task allocation. On the other hand, K-means uses a powerful CPU (40%) while consuming little memory (29%). According to these results, multi-agent is efficient for handling low CPU loads, while K-Means is better suited to computationally-intensive systems.The analytical findings indicate that the selection of the compression method is based not only on performance criteria such as time, size, and readability, but also on the available hardware configuration. This is essential for a multi-criteria evaluation that takes resource constraints into account.

This study aims to highlight methods that can improve storage efficiency in the context of Big Data, which were the subject of this study because of the results obtained. All four algorithms showed high performance in comparison with the identified criteria (compressed size - compression time - read time - average CPU, RAM), although AI-based algorithms have an advantage. Other compression algorithms such as RLE (Run-Length Encoding) and Huffman were used to test these techniques, as were the mathematical algorithms such as PSO and Simulated Annealing. However, the results obtained did not match the algorithms presented in this report, particularly about compressed size and compression ratio, which were high, and temporal compression, which was higher.

Our multi-criteria evaluation offers added value by demonstrating that examining a single indicator (such as compression time) is not sufficient to judge the suitability of a storage method. Thus, this research helps to inform practical decisions regarding the storage and management of Big Data.

VII. Conclusion

One of the biggest challenges on a global scale is how to handle the constant flow of big data. This study discusses some of the key methods used to improve data storage, such as compression, duplication, deduplication, segmentation, and virtualization, with particular consideration given to the challenges associated with contemporary storage technologies. These methods form the basis for contemporary solutions, but they remain insufficient without the incorporation of artificial intelligence and its performance and resource requirements. To overcome these constraints, we have suggested a comparative analysis of four approaches designed to address the challenges associated with data storage and management. All of these methods are based on the principle of storage optimization, which consists of maximizing capacity through algorithms designed to improve the use of storage resources. We propose two techniques based on artificial intelligence (multi-agent systems and K-means clustering), as well as two other approaches, PSNR-SSIM and smart storage optimization. These techniques were implemented on a real dataset consisting of images. The performance of the four algorithms compared to established benchmarks was high, although the algorithms based on artificial intelligence had a slight advantage. These results have several implications. Scientifically speaking, these algorithms are all based on the principle of storage optimization, which aims to improve capacity through algorithms designed to maximize the use of storage resources. The application of storage optimization methods in image databases has proven effective in terms of storage size and processing speed. Once decompressed, the data retains its quality without any loss of information, highlighting the power and efficiency of the algorithms. This research could reveal promising new prospects. Future advances include cutting-edge intelligent algorithms and the incorporation of other innovative methods such as deep learning, reinforcement learning, and quantum optimization. These algorithms can handle various types of data of varying sizes, thanks to a better understanding of information and how it is used. They also facilitate informed decision-making regarding data management and storage, compression, and rapid insertion in an optimization context. This improves the efficiency of existing systems while reducing storage costs, particularly when integrating an intelligent structure into an intelligent environment to develop and optimize storage in the era of big data.

Acknowledgment

This work is supported by National Center for Scientific and Technical Research (CNRST), Rabat, Morocco

References

[1] L. Theodorakopoulos, A. Theodoropoulou, Y. Stamatiou, "A State-of-the-Art Review in Big Data Management Engineering: Real-Life Case Studies, Challenges, and Future Research Directions," eng, Vol. 3, Issue. 3, doi: 10.3390/eng5030068, 2024.

[2] A. Siddiqa, A. Gani, and A. Karim, "Big data storage technologies: a survey," Frontiers of Information Technology & Electronic Engineering, vol. 18, pp. 1040–1070, 2017. doi: 10.1631/fitee.1500441

[3] W. Yang, Y. Qin, and Z. Yang, "A reinforcement learning based data storage and traffic management in information-centric data center networks," Mobile Networks and Applications, vol. 27, 2022. doi: 10.1007/s11036-020-01629-w

[4] J. P. Bharadiya, "A comparative study of business intelligence and artificial intelligence with big data analytics," American Journal of Artificial Intelligence, vol. 7, no. 1, pp. 24–30, 2023. doi: 10.11648/j.ajai.20230701.14

[5] A. Charalambous and N. Dodlek, "Big data, machine learning, and artificial intelligence to advance cancer care: opportunities and challenges," Seminars in Oncology Nursing, vol. 39, 2023. doi: 10.1016/j.soncn.2023.151429

[6] J. Zhang, "Multiple roles of big data platform and artificial intelligence architecture," Applied Mathematics, Modeling and Computer Simulation, pp. 985–963, 2023. doi: 10.3233/atde231039

[7] A. Tara, H. K. Turesson, and N. Natea, "Dynamic storage optimization for communication between ai agents," Future Internet, vol. 16, 2024. doi: 10.3390/fi16080274

[8] S. B. Mohammed, B. Krothapalli, and C. Althati, "Advanced techniques for storage optimization in resource-constrained systems using ai and machine learning," Journal of Science & Technology, vol. 4, no. 1, pp. 89–124, 2023.

[9] P. H. Ahmad and M. Rai, "Analysis of optimization strategies for big data storage management: a study," in Proc. 2023 4th Int. Conf. on Electronics and Sustainable Communication Systems, 2023. doi: 10.1109/icesc57686.2023.10193738

[10] T. Zhang *et al.*, "Data management of scientific applications in a reinforcement learning-based hierarchical storage system," Expert Systems With Applications, vol. 237, pt. B, Mar. 2024. doi: 10.1016/j.eswa.2023.121443

[11] Q. Wang, D. Yu, J. Zhou, and C. Jin, "Data storage optimization model based on improved simulated annealing algorithm," Sustainability, vol. 15, 2023. doi: 10.3390/su15097388

[12] D. Sawitri, "Big data challenges and opportunities in the development of digital technology," Jurnal Info Sains: Informatika dan Sains, vol. 14, no. 2, pp. 215–224, 2024. doi: 10.54209/infosains.v14i02.4619

[13] M. Sais, N. Rafalia, R. Mahdaoui, J. Abouchabaka, "Distributed storage optimization using multi-agent systems in Hadoop", E3S Web of Conferences, doi: 10.1051/e3sconf/202341201091, 2023.

[14] A. S. Pothukuchi, L. V. Kota, and V. Mallikarjunaradhya, "A critical analysis of the challenges and opportunities to optimize storage costs for big data in the cloud," Asian Journal of Multidisciplinary Research & Review, vol. 3, no. 1, pp. 132–144, 2022.

[15] A. O. Adenubi, A. P. Oduroye, and A. Akanni, "Data security in big data: challenges, strategies, and future trends," Int. J. of Research in Education Humanities and Commerce, 2024. doi: 10.37602/ijrehc.2024.5201

[16] L. Wang and Y. Zhang, "Access controls in big data environments," Journal of Information Security, vol. 16, no. 3, pp. 112–128, 2019.

[17] M. Sais, N. Rafalia, R. Mahdaoui, and J. Abouchabaka, "Distributed storage optimization using multi-agent systems in Hadoop," E3S Web of Conferences, vol. 412, 2023. doi: 10.1051/e3sconf/202341201091

[18] L. A. Fitriya, T. W. Purboyo, and A. L. Prasasti, "A review of data compression techniques," International Journal of Applied Engineering Research, vol. 12, no. 19, pp. 8956–89633, 2017.

[19] R. V. Bidwe, S. Mishra, S. Patil, K. Shaw, D. R. Vora, K. Kotecha, and B. Zope, "Deep learning approaches for video compression: A bibliometric analysis," Big Data Cognitive Computing, vol. 6, 2022. doi: 10.3390/bdcc6020044

[20] A. Govardhan and R. Dugyani, "Survey on data replication in cloud systems," Web Intelligence, vol. 22, pp. 83–109, 2024. doi: 10.3233/web-230087

[21] S. Ayoub *et al.*, "Intelligent optimization algorithm based data storage and management: an empirical study," in Proc. 2023 2nd Int. Conf. on Electronics and Renewable Systems (ICEARS), 2023. doi: 10.1109/icears56392.2023.10085038

[22] N. Sharma, A. V. K. Prasad, and V. Kakulapati, "Data deduplication techniques for big data storage systems," Int. J. of Innovative Technology and Exploring Engineering (IJITEE), vol. 8, no. 10, pp. 1145–1150, Aug. 2019. doi: 10.35940/ijitee.j9129.0881019

[23] A. Gulamraza, "Big data virtualization," Int. Research Journal of Modernization in Engineering Technology and Science, vol. 5, no. 4, pp. 5104–5107, 2023.

[24] B. Zohuri and F. M. Rahmani, "Artificial intelligence driven resiliency with machine learning and deep learning components," Journal of Communication and Computer, 2019. doi: 10.17265/1548-7709/2019.01.001

[25] Ch. Cheng, Ch. Zhou, Y. Zhao, J. Cao, "Dynamic Optimization of Storage Systems Using Reinforcement Learning Techniques, "arXiv, 2024.

YOLOv8s-Swin: Enhanced Tomato Ripeness Detection for Smart Agriculture

Jalal Uddin Md Akbar, Syafiq Fauzi Kamarulzaman*
Faculty of Computing, Universiti Malaysia Pahang Al-Sultan Abdullah
Pekan, 26600, Pahang, Malaysia

***Abstract*—Accurate object detection and classification are paramount in precision agriculture for assessing ripeness stages and optimizing yield, particularly for high-value crops like tomatoes. Traditional manual inspection methods are laborious, time-consuming, and error-prone. Furthermore, existing deep learning models often struggle with real-world agricultural challenges such as varying lighting, occlusions from foliage or other fruits, and dense clustering of small objects. To address these limitations and enhance tomato production efficiency and quality in diverse agricultural conditions, this study introduces YOLOv8s-Swin, an advanced object detection model. YOLOv8s-Swin integrates the powerful YOLOv8s architecture with a Swin Transformer module (C3STR) to capture global and local contextual information, crucial for robust small object detection. It also incorporates Focus, Depthwise Convolution (DWconv), Spatial Pyramid Pooling with Contextual Spatial Pyramid Convolution (SPPCSPC), and C2 modules for preserving fine details, reducing computational overhead, enhancing multi-scale feature fusion, and improving high-level semantic feature extraction, respectively. The Wise Intersection over Union (WIoU) loss function is adopted to enhance localization and address convergence issues. Evaluated on a comprehensive tomato image dataset, YOLOv8s-Swin demonstrated superior performance with a mean Average Precision (mAP@0.5) of 88.3%, precision of 84.4%, recall of 79.9%, and an F1-Score of 0.821. This significantly surpasses the base YOLOv8s (84.7% mAP@0.5, 0.795 F1-Score) and other models like Faster R-CNN, SSD, YOLOv4, YOLOv5s, and YOLOv7, all under identical conditions. Maintaining a competitive inference speed of 166.67 FPS, YOLOv8s-Swin offers a robust and efficient solution for AI-driven crop management and sustainable food production.**

***Keywords*—Agricultural automation; attention mechanism; computer vision; smart agriculture; object detection; YOLO; swin transformer**

I. Introduction

The agricultural sector is undergoing a profound transformation driven by advancements in computer vision (CV) and deep learning algorithms. Accurate object detection, a key facet of CV, plays a pivotal role in modern precision agriculture, enabling automated tasks such as crop variety identification and ripeness assessment. For high-value crops like tomatoes, optimizing yield and quality within dynamic agricultural environments necessitates precise and timely detection and classification of their ripeness levels [1]. Traditional manual inspection methods are laborious, time-consuming, and error-prone. Convolutional Neural Networks (CNNs) have spearheaded significant progress in this domain, providing robust solutions for extracting hierarchical visual features from images [2]. Among the prominent deep learning architectures, the You Only Look Once (YOLO) series has revolutionized object detection by achieving remarkable speed and accuracy in a single forward pass [3]. Each iteration of the YOLO architecture strives to push the boundaries of both computational efficiency and detection performance. This study focuses on leveraging and enhancing the YOLOv8 architecture, known for its balance of speed and precision, particularly in detecting smaller objects [4].

Despite the efficacy of current deep learning models, challenges persist in real-world agricultural settings. Factors such as varying lighting conditions, occlusions from foliage or other fruits, and dense clustering can significantly impact detection accuracy [5]. Small objects, in particular, often lack sufficient semantic information due to limited pixels, making their accurate detection difficult [6]. To address these challenges, researchers have increasingly explored the integration of attention mechanisms and advanced architectural components into object detection models. Attention mechanisms allow models to selectively focus on the most relevant features in an image, improving feature representation and reducing the influence of irrelevant background noise [7], [8].

Inspired by the success of transformer-based architectures in capturing long-range dependencies and global contextual information in natural language processing and, more recently, in computer vision [9], this study proposes YOLOv8s-Swin, an enhanced object detection model for tomato ripeness detection in smart agriculture. Previous versions of YOLO, including YOLOv4 and YOLOv5s, have shown strong performance in object detection but face significant challenges when it comes to detecting small, occluded, or densely clustered objects in complex environments [25]. These models often struggle with contextual awareness, which is critical for distinguishing between closely packed fruits or handling occlusions from foliage. While YOLOv8s introduced improvements in detection speed and accuracy, particularly for small object detection, it still shares similar limitations to its predecessors. The base YOLOv8s model's focus on local feature extraction and its relatively shallow architectural layers hinder its ability to capture broader contextual information needed for accurate detection in agricultural environments [26]. The model's performance drops when faced with the dense clustering of tomatoes or partial occlusions by other fruits or leaves. To address these issues, YOLOv8s-Swin integrates the Swin Transformer module, which allows for better capture of both local and global context through its self-attention mechanism. This integration, along with additional architectural improvements such as the Focus, Depthwise Convolution (DWconv), SPPCSPC, and C2 modules, enhances the model's ability to handle small and occluded objects while improving feature extraction, computational efficiency, and multi-scale feature

fusion. Thus, YOLOv8s-Swin overcomes the limitations of the base YOLOv8s model, providing a more robust and efficient solution for detecting tomatoes at various ripeness stages under diverse agricultural conditions.

The primary contributions of this study are:

- Development of YOLOv8s-Swin, an enhanced architecture for robust tomato ripeness detection. This model integrates the Swin Transformer (C3STR), Focus, DWconv, SPPCSPC, C2 modules, and the WIoU loss function, optimizing feature extraction and localization for challenging agricultural scenes.
- Demonstration of superior performance in classifying tomato ripeness across diverse agricultural conditions. YOLOv8s-Swin achieved an 88.3% mAP@0.5, significantly outperforming the base YOLOv8s (84.7%) and other models, effectively handling occlusions and lighting variations.
- Achieving high accuracy while maintaining real-time inference capabilities. With an FPS of 166.67, YOLOv8s-Swin offers a practical and efficient solution for AI-driven crop management and sustainable food production.

The remainder of this study is organized as follows: Section II provides a comprehensive review of relevant literature on object detection, with a focus on advanced architectures. Section III details the materials and methods used, including data acquisition, preprocessing, and the architectural specifics of the proposed YOLOv8s-Swin model, as well as the experimental setup. Section IV presents and discusses the quantitative and qualitative results, including a comparative analysis with baseline models. Finally, Section V concludes the study by summarizing key findings and outlining directions for future research.

II. Related Work

Object detection has been a cornerstone of computer vision for decades. Early approaches, such as two-stage detectors like R-CNN [10], Fast R-CNN [11], and Faster R-CNN [12], achieved high accuracy by first generating region proposals and then classifying and refining them. While accurate, these methods often struggled with real-time applications due to their computational intensity.

The advent of one-stage detectors, notably the YOLO (You Only Look Once) series, marked a significant shift by performing object localization and classification in a single forward pass, dramatically increasing detection speed [3]. Subsequent YOLO versions, including YOLOv2 [13], YOLOv3 [14], YOLOv5 [15], YOLOv7 [16], and YOLOv8 [4], have continuously pushed the boundaries of speed and accuracy, making them highly suitable for real-time applications. YOLOv8, in particular, introduced advancements in its backbone network and decoupled heads, enhancing its performance, especially for smaller objects.

Small object detection remains a challenging problem due to the limited pixel information available for feature extraction and the lack of sufficient contextual background [6], [27]. To address this, various strategies have been proposed. Attention mechanisms have also gained prominence, allowing models to focus on important features and suppress irrelevant ones, thereby improving representation [7], [8]. Channel attention (e.g., Squeeze-and-Excitation Networks [17]) and spatial attention (e.g., Convolutional Block Attention Module (CBAM) [18]) are widely used variants. Recent works, such as those by Chien et al. [19] and Najihah Muhamad Zamri et al. [20], have demonstrated the benefits of integrating attention mechanisms into YOLO models for improved detection in various applications.

The introduction of Transformer architectures, initially successful in natural language processing, has significantly impacted computer vision. The Swin Transformer [9] is particularly relevant for its hierarchical vision transformer using shifted windows, enabling it to capture global and local contextual information efficiently while maintaining computational feasibility[28]. This makes it a strong candidate for enhancing object detection models, especially for challenging scenarios like small object detection in complex environments. Shi et al. [6] successfully integrated a Swin Transformer module into a YOLOv8 network (STF-YOLO) for small object detection of tea buds, demonstrating significant improvements in accuracy.

In the context of crop ripeness detection, several studies have leveraged YOLO-based models. Li et al. [21] used YOLOv5 for tomato maturity detection in greenhouses, achieving high precision but facing potential limitations in outdoor conditions. Li et al. [22] proposed MHSA-YOLOv8 for tomato grading and counting, which performed well in complex environments but struggled with severe occlusion. Lightweight approaches like those by Zeng et al. [23] and Su et al. [24] emphasized real-time performance but sometimes faltered with dense small targets. While these studies show the promise of YOLO in agriculture, the specific challenges of dense fruit clusters, occlusions, and varied lighting in tomato detection within smart agriculture, combined with the need for robust small object detection, highlight the necessity for further architectural enhancements. Our proposed YOLOv8s-Swin aims to bridge this gap by combining the strengths of YOLOv8s with the contextual awareness of the Swin Transformer and other efficient modules to improve tomato ripeness detection.

III. Materials and Methods

This section details the proposed YOLOv8s-Swin model, its integration of key components, and the comprehensive methodological workflow for enhanced tomato ripeness detection in smart agriculture.

A. Data Acquisition

For this research, the Laboro Tomato dataset (https://github.com/laboroai/LaboroTomato) was utilized. This dataset contains images of tomatoes at three ripeness stages (fully-ripened, half-ripened, and green), reflecting diverse agricultural conditions and camera variations. As shown in Fig. 1, fully ripened tomatoes exhibit >90% red, half-ripened 30-89% red, and green 0-30% red coloration, crucial for accurate ripeness classification.

B. Data Pre-processing and Preparation

Raw images underwent rigorous preprocessing: auto-orientation, uniform resizing to 640×640 pixels, class modification (see Table I), and null annotation filtering. The dataset was split into training (92%, 2001 images), validation (4%, 86 images), and test (4%, 87 images) sets. Extensive data augmentation via Roboflow (see Fig. 2) was applied, including random flips, rotations, shears, blur, and noise, to simulate diverse agricultural conditions and enhance generalization.

TABLE I. REMAPPING OF ORIGINAL CLASSES TO SIMPLIFIED CLASSES IN THE DATASET

Original Class	Override	Included
b_fully_ripened	fully_ripened	Yes
b_green	green	Yes
b_half_ripened	half_ripened	Yes
l_fully_ripened	fully_ripened	Yes
l_green	green	Yes
l_half_ripened	half_ripened	Yes



Fig. 1. Dataset class.

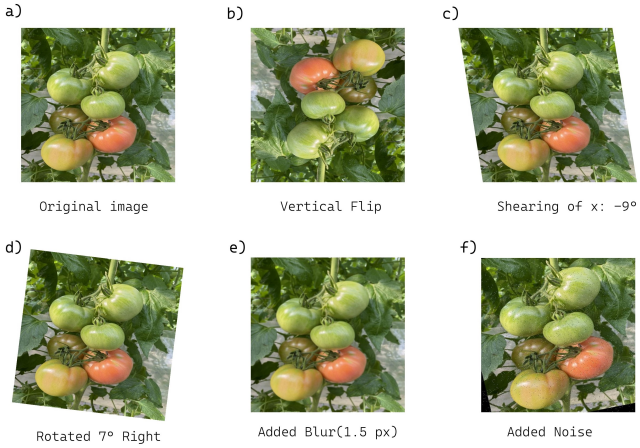


Fig. 2. Data augmentation techniques applied to tomato images; (a) Original image, (b) Vertical Flip, (c) Shearing of X: -9°, (d) Rotated 7° Right, (e) Added Blur (1.5px), (f) Added Noise.

C. Proposed Method: YOLOv8s-Swin Architecture

YOLOv8s-Swin is an advanced object detection architecture building on YOLOv8s, integrating a Swin Transformer module and other components. This enhances focus and contextual information capture, improving accuracy for small and occluded objects. The architecture follows a typical Backbone, Neck, and Head structure, with modifications to feature processing (see Fig. 3).

1) Swin Transformer Module (C3STR): The core innovation is the C3STR module (see Fig. 4), a Swin Transformer integration that mitigates feature loss in deep networks by establishing global dependencies via self-attention [9]. Integrated into the YOLOv8s backbone and neck, C3STR enhances semantic information and representation for small objects. It comprises Window/Shifted Window Multi-Headed Self-Attention (W-MSA/SW-MSA) and Multi-Layer Perceptron (MLP) with internal residual connections. Its self-attention mechanism is:

$$\text{Attention}(Q, K, V) = \text{SoftMax}\left(\frac{QK^T}{\sqrt{d}} + B\right)V \quad (1)$$

C3STR controls computations within local windows, enabling cross-window information interaction while reducing complexity, allowing adaptive feature interaction and capturing crucial contextual information.

2) Additional Modules: Focus, DWconv, SPPCSPC, and C2: To optimize YOLOv8s-Swin, efficient modules are incorporated:

- Focus Module: Placed at the input, it increases channel dimensions while reducing spatial ones, preserving fine details for small objects.
- Depthwise Convolution (DWconv): Performs independent channel convolutions, reducing parameters and computation while maintaining high performance.
- Spatial Pyramid Pooling with Contextual Spatial Pyramid Convolution (SPPCSPC): Captures multi-scale feature information efficiently via pooling and convolution.
- C2 Module: A refined C3 version for enhanced high-level semantic feature extraction with improved memory and inference speed.

These modules, integrated into the YOLOv8s backbone and neck, enable accurate tomato ripeness detection in complex agricultural conditions.

3) Loss Function (WIoU): To address YOLOv8's convergence issues and enhance localization, the Wise Intersection over Union (WIoU) loss function is adopted. Unlike traditional IoU, WIoU considers positional relationships and geometric factors, mitigating the negative impact of low-quality examples and reducing centroid distance emphasis for well-overlapping frames. The WIoUv1 model's calculation is:

$$L_{WIOU \cdot wl} = R_{WIoU} L_{IOU} \quad (2)$$

$$R_{WIoU} = \exp\left(\frac{(x - x_{gl})^2 + (y - y_{gl})^2}{(W_g^2 + H_g^2)^*}\right) \quad (3)$$

$$L_{IOU} = 1 - IOU = 1 - \frac{W_i H_i}{S_u} \quad (4)$$

Here, $R_{\text{WIoU}} \in [1, e)$ amplifies L_{IoU} for normal anchor frames but reduces it for high-quality ones, focusing learning

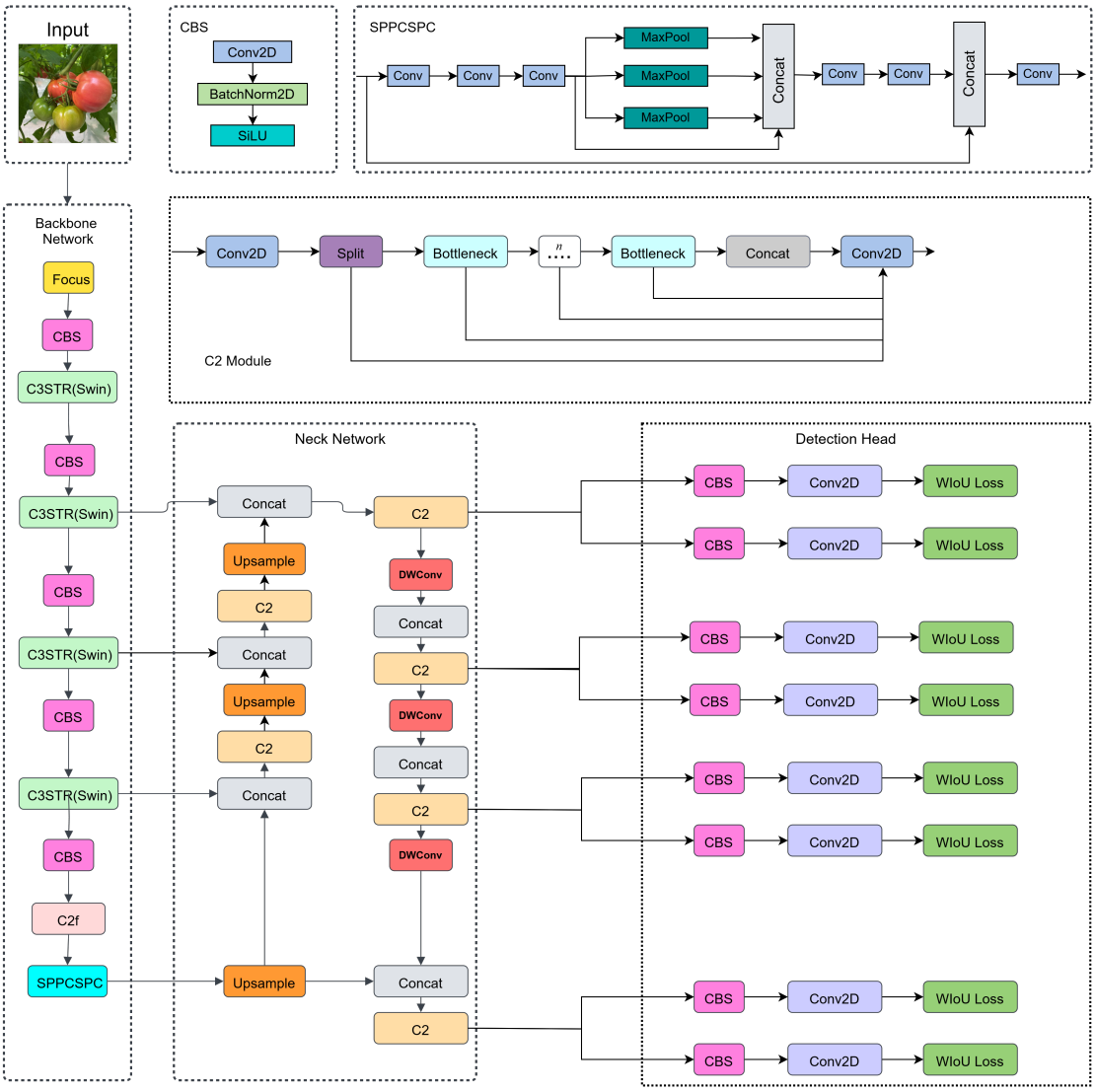


Fig. 3. The architecture of YOLOv8s-Swin for tomato ripeness detection.

on better examples for improved generalization and stability. S_u is the union area.

D. Experimental Setup

Experiments were conducted on an NVIDIA GeForce RTX 3060 GPU (16 GB VRAM) with an AMD Ryzen 5 processor, running Ubuntu 24.04 LTS and PyTorch 2.3.0. Training used 640×640 pixel images with a batch size of 8. Key hyperparameters included: 150 epochs, initial learning rate of 0.01, momentum 0.937, weight decay 0.0005, and data augmentation parameters (mixup 0.15, copy paste 0.3).

IV. Results and Discussion

A. Evaluation Metrics

The performance of the YOLOv8s-Swin model for tomato ripeness detection was quantitatively assessed using several standard object detection metrics:

1) Precision (P): The proportion of correctly identified positive samples among all detected positives. The precision metric is calculated using the following formula:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{5}$$

In this context, TP refers to true positives, representing instances that were correctly identified as positive. Conversely, FP refers to false positives, which are instances incorrectly classified as positive. Attaining a higher precision value signifies a reduction in false alarms within the detection process, leading to a more reliable model.

2) Recall (R): The ratio of correctly identified positive samples to the total number of actual positive samples. The recall metric is defined mathematically as:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{6}$$

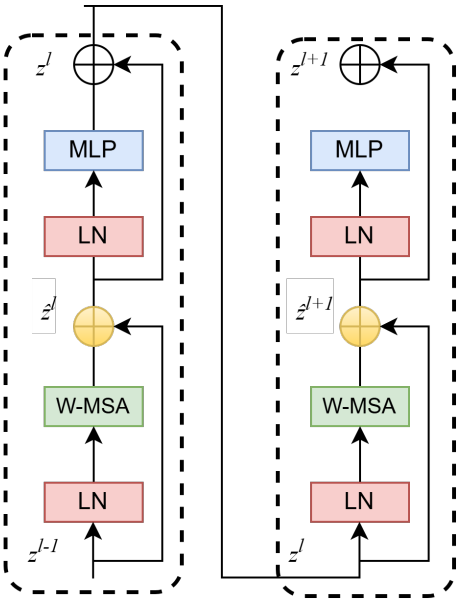


Fig. 4. C3STR architecture diagram.

Here, TP corresponds to true positives, referring to correctly identified positive cases. FN, on the other hand, stands for false negatives, which are genuine positive instances that the model overlooked. Achieving a higher recall value indicates a reduction in missed relevant instances, thereby ensuring a more complete detection capability.

3) mAP (mean average precision): Calculated at an Intersection over Union (IoU) threshold of 0.5 (mAP@0.5) and also across multiple IoU thresholds from 0.5 to 0.95 (mAP@0.5:0.95). It represents the average of the Average Precision (AP) for all classes, derived from the area under the precision-recall curve.

4) F1-Score: The harmonic mean of precision and recall, providing a balanced measure of the model's accuracy.

5) FPS (Frames Per Second): Measures the number of images the model can process per second at a batch size of 1, indicating the model's detection speed.

B. Performance of YOLOv8s-Swin

The proposed YOLOv8s-Swin model was rigorously evaluated on the Laboro Tomato dataset, encompassing fully ripe, partially ripe, and unripe tomatoes in diverse agricultural environments. The model demonstrated strong performance across all evaluation metrics, showcasing its effectiveness in accurately detecting and classifying tomato ripeness stages.

Our YOLOv8s-Swin model achieved an impressive mAP@0.5 of 88.3%, a precision of 84.4%, and a recall of 79.9%. The F1-Score for the model was 0.821. For individual classes, the precision-recall curve (see Fig. 5) highlights mAP@0.5 values of 0.893 for 'fully_ripened', 0.882 for 'green', and 0.873 for 'half_ripened' tomatoes, indicating consistent performance across ripeness stages. The F1-Confidence Curve (see Fig. 6) shows an optimal F1-Score of 0.84 for all classes at a confidence threshold of 0.456.

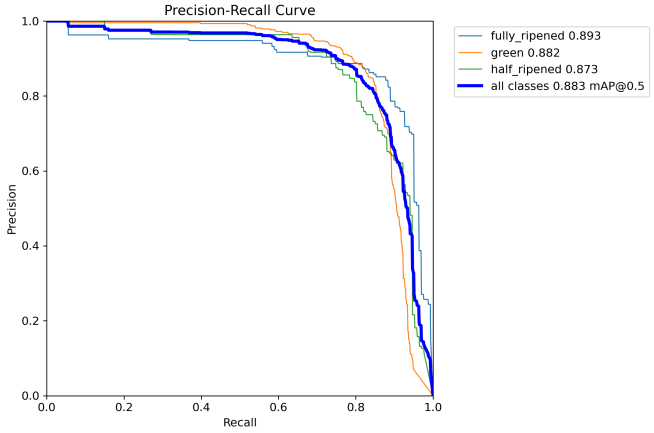


Fig. 5. Precision-recall curve of proposed YOLOv8s-Swin.

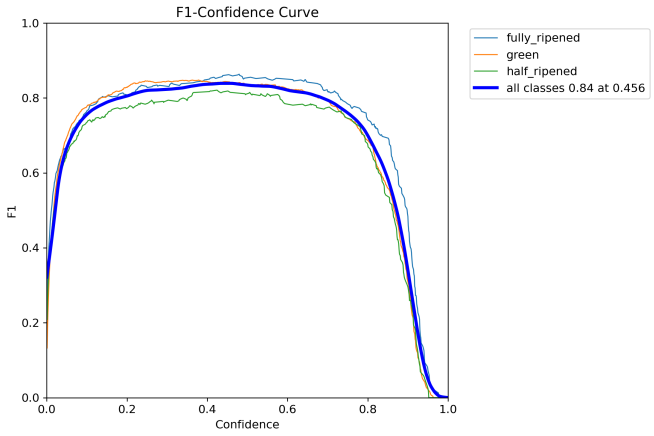


Fig. 6. F1-Confidence curve of proposed YOLOv8s-Swin.

The confusion matrix (see Fig. 7) further confirms the model's strong classification ability. It shows high true positive counts for 'fully_ripened' (144), 'green' (476), and 'half_ripened' (134) predictions, indicating a low rate of misclassification. Furthermore, the training and validation loss curves for bounding box regression, classification, and the overall loss (see Fig. 8) exhibit a consistent and steady decline over training epochs, demonstrating effective learning and robust generalization to unseen data.

C. Comparative Performance Analysis

To thoroughly assess the efficacy and advantages of our proposed YOLOv8s-Swin algorithm, a comprehensive comparative analysis was conducted against several prominent object detection models. These models include traditional two-stage detectors like Faster R-CNN, one-stage detectors such as SSD, and various iterations of the YOLO series, namely YOLOv4, YOLOv5s, YOLOv7, and the base YOLOv8s model. Crucially, all comparative experiments were performed under strictly controlled and identical conditions, utilizing the same hardware setup (NVIDIA GeForce RTX 3060 GPU), the Laboro Tomato dataset, consistent data augmentation methodologies, and a consistent training regimen of 150 epochs, with the optimal

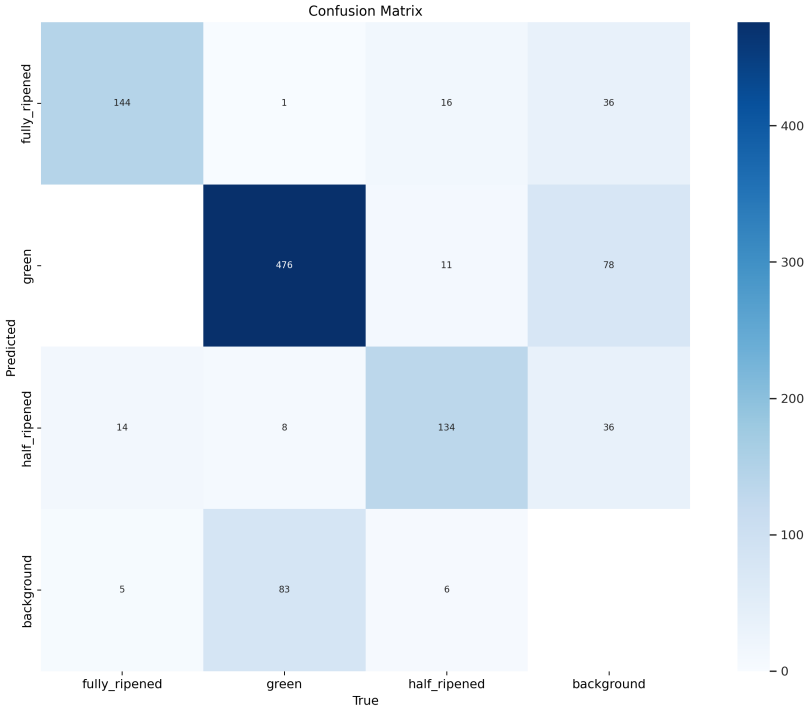


Fig. 7. Confusion matrix of proposed YOLOv8s-Swin.

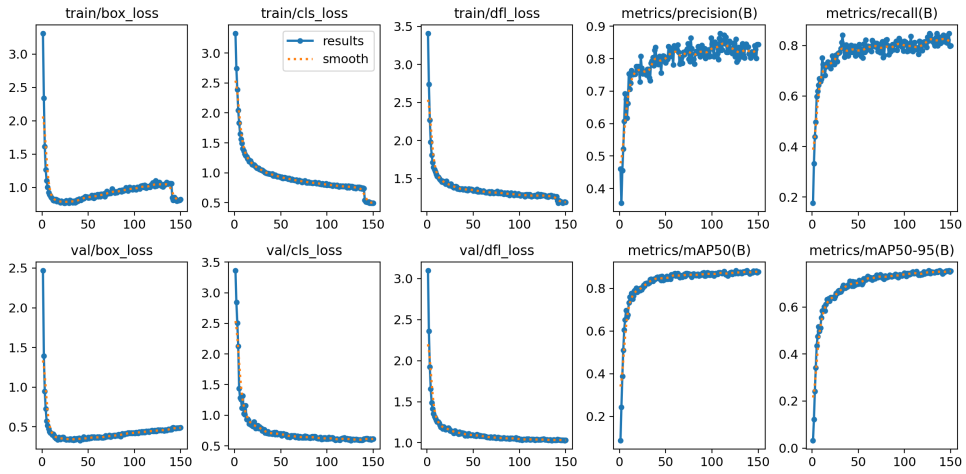


Fig. 8. Visualization of different loss curves of proposed YOLOv8s-Swin model over epochs.

results selected for evaluation.

As presented in Table II, the proposed YOLOv8s-Swin model demonstrates significant improvements across key performance metrics, particularly when directly compared to its base counterpart, YOLOv8s. YOLOv8s-Swin achieved a notably higher mAP@0.5 of 88.3% compared to the base YOLOv8s's 84.7%, representing a substantial improvement in overall detection accuracy under identical experimental settings. Furthermore, YOLOv8s-Swin's precision is 84.4%, an increase from YOLOv8s's 78.3%, indicating its superior accuracy in identifying true positive instances. The recall for YOLOv8s-Swin also saw an improvement, reaching 79.9% compared to YOLOv8s's 78.2%, meaning it is more effective at capturing a higher proportion of actual positive cases. The F1-Score, a crucial balanced measure of precision and recall, also significantly improved from 0.795 for YOLOv8s to 0.821 for YOLOv8s-Swin. These advancements in detection accuracy, precision, recall, and F1-Score are a direct result of the integrated architectural enhancements.

The superior performance of YOLOv8s-Swin is primarily due to the strategic integration of several advanced modules into the YOLOv8 framework. Specifically, the Swin Transformer module significantly enhances the model's ability to capture global and local contextual information, which is

TABLE II. COMPREHENSIVE MODEL COMPARISON FOR TOMATO RIPENESS DETECTION

Network	P (%)	R (%)	mAP@0.5 (%)	F1-Score	FPS
Faster R-CNN	82.0	78.0	86.0	0.780	10
SSD	78.6	75.0	78.0	0.740	40
YOLOv4	74.0	70.0	76.0	0.719	55
YOLOv5s	78.0	73.5	82.0	0.762	110
YOLOv7	78.3	75.0	83.0	0.779	120
YOLOv8s	78.3	78.2	84.7	0.795	160
YOLOv8s-Swin (Proposed)	**84.4**	**79.9**	**88.3**	**0.821**	**166.67**

crucial for accurately detecting small and partially occluded tomatoes in complex agricultural scenes. Additionally, the inclusion of the Focus module aids in preserving fine-grained details vital for small objects, Depthwise Convolution (DWconv) reduces computational overhead while maintaining feature richness, SPPCSPC enhances multi-scale feature fusion, and the C2 module improves high-level semantic feature extraction. These combined modules, along with the robust WIoU loss function, ensure that critical information is effectively utilized during the learning process, leading to more reliable and robust feature representations, particularly beneficial for accurately recognizing subtle differences between various stages of tomato maturity.

Beyond the direct comparison with YOLOv8s, YOLOv8s-Swin exhibits remarkable performance against all other models tested under identical experimental settings. It achieves the highest mAP@0.5 of 88.3%, significantly surpassing traditional two-stage detectors like Faster R-CNN (86.0%) and SSD (78.0%), as well as earlier YOLO iterations such as YOLOv4 (76.0%), YOLOv5s (82.0%), and YOLOv7 (85.0%). The F1-Score of 0.821 is also the highest among all compared models, underscoring its superior balance between precision and recall for robust real-world performance.

In terms of detection speed, YOLOv8s-Swin maintains a highly competitive FPS rate of 166.67, which is marginally higher than the base YOLOv8s model's 160 FPS. This satisfies the stringent requirements for real-time detection in agricultural applications. In contrast, older models like Faster R-CNN (10 FPS) and SSD (40 FPS) fall significantly short of real-time capabilities. This critical balance between high accuracy and efficient processing makes YOLOv8s-Swin a highly universal and practical solution for smart agriculture, particularly suitable for deployment on edge devices with limited processing capabilities without substantial hardware investments.

D. Qualitative Analysis

To visually demonstrate the effectiveness of the proposed YOLOv8s-Swin model, Fig. 9 presents several detection examples across various agricultural conditions, including instances with varying lighting, occlusions, and different ripeness stages. The model consistently exhibits strong performance in identifying and classifying tomatoes at their respective ripeness levels. For instance, Fig. 9(a), Fig. 9(b), and Fig. 9(f) clearly illustrate the model's capability to accurately detect and categorize green (unripe) tomatoes, even when they are densely clustered or partially obscured by surrounding foliage. This highlights the model's resilience to common real-world challenges in agricultural settings.

Furthermore, Fig. 9(c) and Fig. 9(e) showcase the precise detection of fully-ripened tomatoes, with the bounding boxes and associated confidence scores, indicating a high degree of accuracy in classification. This performance is crucial for timely harvesting and yield optimization. Fig. 9(d) specifically demonstrates the model's nuanced ability to differentiate between half-ripened and fully-ripened tomatoes, underscoring its capacity to discern subtle visual cues that distinguish various stages of maturity. Collectively, these qualitative results visually confirm the robustness and practical applicability of the YOLOv8s-Swin model in diverse and challenging agricultural environments, reinforcing its strong quantitative performance metrics through accurate bounding box localization and precise ripeness classification under real-world conditions.

E. Discussion

Accurate detection and classification of crop ripeness stages are indispensable for informed decision-making and optimizing yield management in precision agriculture. Our proposed YOLOv8s-Swin model significantly advances the state-of-the-art for tomato ripeness detection, demonstrating a new benchmark for object detection and classification in agricultural applications, specifically for tomatoes.

The superior performance of YOLOv8s-Swin, particularly its notably higher mAP@0.5, precision, recall, and F1-Score compared to baseline models like YOLOv8s and other prominent architectures, is primarily attributable to its innovative architectural enhancements. The strategic integration of the Swin Transformer module (C3STR) proved crucial, allowing the model to better capture global and local contextual information. This enhanced contextual awareness is particularly beneficial for accurately detecting small and partially occluded tomatoes in complex agricultural scenes. Furthermore, the inclusion of the Focus module aids in preserving fine-grained details, while Depthwise Convolution (DWconv) reduces computational overhead without sacrificing feature richness. The SPPCSPC module enhances multi-scale feature fusion, and the C2 module improves high-level semantic feature extraction by being a refined version of C3. These combined architectural modifications, coupled with the robust Wise Intersection over Union (WIoU) loss function, which mitigates the negative impact of low-quality examples and enhances localization, ensure that critical information is effectively utilized during the learning process. This leads to more reliable and robust feature representations, particularly beneficial for accurately recog-

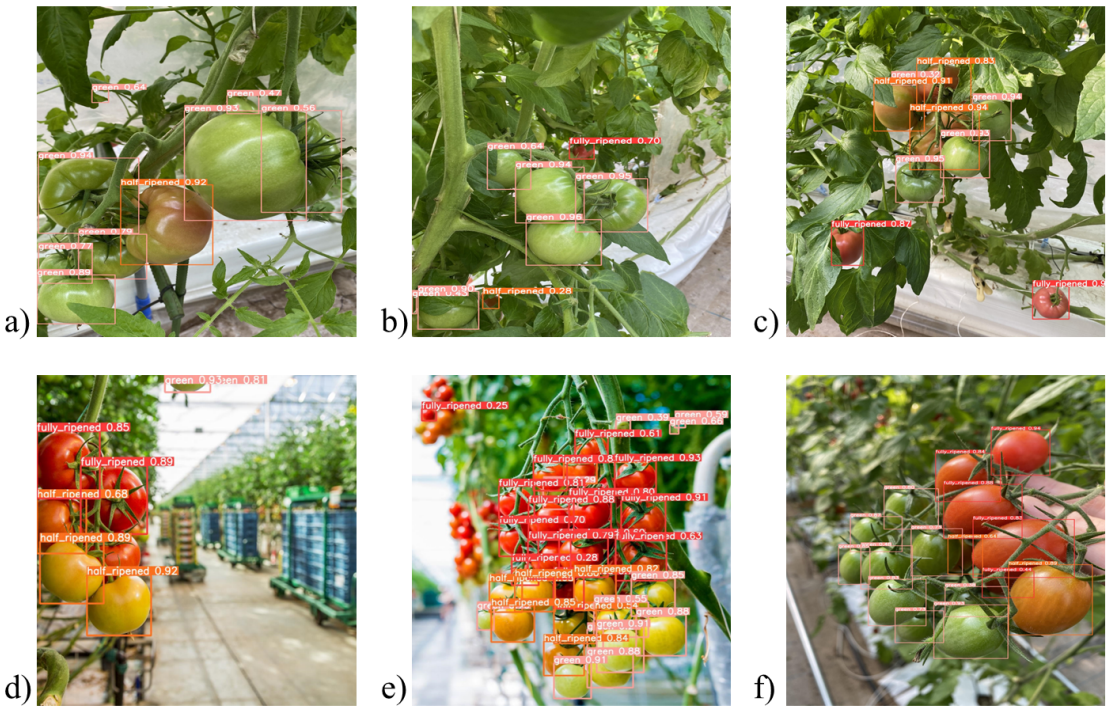


Fig. 9. YOLOv8s-Swin's robust detection capabilities across diverse smart agriculture conditions, including: (a) fruits occluded by other fruits, (b) fruits occluded by foliage, (c) combined occlusion by foliage and fruits, (d) detection in high-density fruit environments, (e) accurate detection in densely packed fruits, and (f) high-density fruit clustering detection.

nizing subtle differences between various stages of tomato maturity.

Beyond its significant accuracy improvements, YOLOv8s-Swin maintains a highly competitive inference speed of 166.67 FPS, which is practical for real-time applications in agriculture. This speed, combined with the model's moderate computational requirements, facilitates efficient deployment on edge devices commonly found in agricultural settings, thereby minimizing the need for substantial hardware investments. The model's adaptability also suggests potential for customization across various platforms, enhancing scalability for diverse agricultural operations.

Despite the high accuracy achieved, certain challenges persist. The model may still struggle with severely occluded tomatoes or distinguishing them from background foliage in extremely dense clusters. Future research will focus on several key areas to build upon these findings. Integrating multi-modal data sources, such as infrared or depth images, could significantly enhance the model's robustness against challenging occlusions and lighting variations. Furthermore, exploring advanced optimization techniques, including neural architecture search, could lead to even more tailored and efficient designs for this specific task. Deployment-specific optimizations, such as quantization-aware training, will be pursued to further reduce computational requirements, enabling broader and more efficient use in resource-constrained environments.

V. Conclusion

This study successfully introduced YOLOv8s-Swin, an enhanced object detection model for robust tomato ripeness detection within smart agriculture. A primary contribution lies in the innovative integration of the Swin Transformer module, Focus, Depthwise Convolution (DWconv), Spatial Pyramid Pooling with Contextual Spatial Pyramid Convolution (SPPCSPC), and C2 modules into the YOLOv8s framework. These architectural enhancements proved highly effective, significantly improving the model's ability to detect small, occluded objects and handle complex agricultural environments. The model achieved a mean Average Precision (mAP@0.5) of 88.3%, precision of 84.4%, recall of 79.9%, and an F1-Score of 0.821, surpassing the base YOLOv8s (84.7% mAP@0.5, 78.3% precision, 78.2% recall, 0.795 F1-Score) and other models like Faster R-CNN, SSD, YOLOv4, YOLOv5s, and YOLOv7, all trained under identical conditions. These improvements are attributed to the model's ability to capture both local and global context, addressing the limitations of previous YOLO versions, which struggled with occlusions, lighting variations, and dense clustering of objects. Additionally, YOLOv8s-Swin maintains a competitive inference speed of 166.67 FPS, demonstrating its applicability in real-time applications for AI-driven crop management and sustainable food production. While the model has demonstrated impressive performance, future work will focus on optimizing it for deployment in more resource-constrained environments, integrating multimodal data sources like infrared and depth sensing to further enhance accuracy under challenging conditions, and exploring the applicability of the model to other crops. The advancements presented

here significantly contribute to the field of smart agriculture by providing an efficient and scalable solution for detecting ripeness stages in tomatoes, paving the way for future research and development in AI-driven agricultural automation.

ACKNOWLEDGMENT

We sincerely thank the UMPSA Postgraduate Research Grants Scheme (PGRS) for their support through the Open Access funding, which made this research possible.

REFERENCES

[1] M. Rizzo, M. Marcuzzo, A. Zangari, A. Gasparetto, and A. Albarelli, "Fruit ripeness classification: A survey," *Artificial Intelligence in Agriculture*, vol. 7, pp. 44–57, 2023. doi: 10.1016/j.aiia.2023.02.004.

[2] J. U. Md. Akbar, S. F. Kamarulzaman, A. J. Md. Muzahid, M. A. Rahman, and M. Uddin, "A Comprehensive Review on Deep Learning Assisted Computer Vision Techniques for Smart Greenhouse Agriculture," *IEEE Access*, vol. 12, pp. 4485–4522, 2024. doi: 10.1109/ACCESS.2024.3349418.

[3] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, Jun. 2016.

[4] S. Ma, H. Lu, J. Liu, Y. Zhu and P. Sang, "LAYN: Lightweight Multi-Scale Attention YOLOv8 Network for Small Object Detection," in IEEE Access, vol. 12, pp. 29294-29307, 2024, doi: 10.1109/ACCESS.2024.3368848.

[5] R. Li, Z. Ji, S. Hu, X. Huang, J. Yang, and W. Li, "Tomato Maturity Recognition Model Based on Improved YOLOv5 in Greenhouse," *Agronomy*, vol. 13, no. 2, Art. no. 603, 2023. doi: 10.3390/agronomy13020603.

[6] M. Shi, D. Zheng, T. Wu, W. Zhang, R. Fu, and K. Huang, "Small object detection algorithm incorporating swin transformer for tea buds," *PLoS ONE*, vol. 19, no. 3, pp. 1–25, 2024. doi: 10.1371/journal.pone.0299902.

[7] M.-H. Guo *et al.*, "Attention mechanisms in computer vision: A survey," *Computational Visual Media*, vol. 8, no. 3, pp. 331-368, 2022. doi: 10.1007/s41095-022-0271-y.

[8] H. Yao, Y. Liu, X. Li, Z. You, Y. Feng, and W. Lu, "A Detection Method for Pavement Cracks Combining Object Detection and Attention Mechanism," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 22179–22189, 2022. doi: 10.1109/TITS.2022.3177210.

[9] Z. Liu *et al.*, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 10012-10022.

[10] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 580-587.

[11] J. Li, X. Liang, S. Shen, T. Xu, J. Feng and S. Yan, "Scale-Aware Fast R-CNN for Pedestrian Detection," in *IEEE Transactions on Multimedia*, vol. 20, no. 4, pp. 985-996, April 2018, doi: 10.1109/TMM.2017.2759508.

[12] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017. doi: 10.1109/TPAMI.2016.2577031.

[13] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, Jul. 2017.

[14] Q. -C. Mao, H. -M. Sun, Y. -B. Liu and R. -S. Jia, "Mini-YOLOv3: Real-Time Object Detector for Embedded Applications," in IEEE Access, vol. 7, pp. 133529-133538, 2019, doi: 10.1109/ACCESS.2019.2941547.

[15] C. Wang *et al.*, "A Lightweight Cherry Tomato Maturity Real-Time Detection Algorithm Based on Improved YOLOV5n," *Agronomy*, vol. 13, no. 8, Art. no. 2106, 2023. doi: 10.3390/agronomy13082106.

[16] Y. Wu *et al.*, "An improved YOLOv7 network using RGB-D multi-modal feature fusion for tea shoots detection," *Computers and Electronics in Agriculture*, vol. 216, pp. 108541, 2024. doi: 10.1016/j.compag.2023.108541.

[17] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation Networks," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, Jun. 2018.

[18] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional Block Attention Module," in *Proc. Eur. Conf. Computer Vision (ECCV)*, Munich, Germany, Sep. 2018.

[19] C. -T. Chien, R. -Y. Ju, K. -Y. Chou, E. Xieerke and J. -S. Chiang, "YOLOv8-AM: YOLOv8 Based on Effective Attention Mechanisms for Pediatric Wrist Fracture Detection," in *IEEE Access*, vol. 13, pp. 52461-52477, 2025, doi: 10.1109/ACCESS.2025.3549839.

[20] F. N. M. Zamri *et al.*, "Enhanced Small Drone Detection Using Optimized YOLOv8 With Attention Mechanisms," *IEEE Access*, vol. 12, pp. 90629–90643, 2024. doi: 10.1109/ACCESS.2024.3420730.

[21] R. Li *et al.*, "Tomato Maturity Recognition Model Based on Improved YOLOv5 in Greenhouse," *Agronomy*, vol. 13, no. 2, Art. no. 603, 2023. doi: 10.3390/agronomy13020603.

[22] P. Li *et al.*, "Tomato Maturity Detection and Counting Model Based on MHSA-YOLOv8," *Sensors*, vol. 23, no. 15, Art. no. 6701, 2023. doi: 10.3390/s23156701.

[23] T. Zeng, S. Li, Q. Song, F. Zhong, and X. Wei, "Lightweight tomato real-time detection method based on improved YOLO and mobile deployment," *Computers and Electronics in Agriculture*, vol. 205, pp. 107625, 2023. doi: 10.1016/j.compag.2023.107625.

[24] F. Su *et al.*, "Tomato Maturity Classification Based on SE-YOLOv3-MobileNetV1 Network under Nature Greenhouse Environment," *Agronomy*, vol. 12, no. 7, Art. no. 1638, 2022. doi: 10.3390/agronomy12071638.

[25] Y. Gao, Z. Li, Y. Wang, and S. Zhu, "A Novel YOLOv5_ES based on lightweight small object detection head for PCB surface defect detection," *Scientific Reports*, vol. 14, no. 1, Oct. 2024, doi: https://doi.org/10.1038/s41598-024-74368-7.

[26] L. Shen, B. Lang and Z. Song, "DS-YOLOv8-Based Object Detection Method for Remote Sensing Images," in *IEEE Access*, vol. 11, pp. 125122-125137, 2023, doi: 10.1109/ACCESS.2023.3330844.

[27] J. U. M. Akbar, S. Fauzi Kamarulzaman and E. H. Tusher, "Plant Stem Disease Detection Using Machine Learning Approaches*," *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Delhi, India, 2023, pp. 1-8, doi: 10.1109/ICCCNT56998.2023.10307074.

[28] J. Jumadi and J. U. M. Akbar, "Hybrid GRU-KAN model for energy consumption prediction in commercial building cooling," International Journal of Advanced Computing and Informatics(IJACI), vol. 1, no. 2, pp. 69–78, Jun. 2025, doi: 10.71129/ijaci.v1.i2.pp69-78.

Next-Generation Network Security: An Analysis of Threats, Challenges and Emerging Intelligent Defenses Within SDN and NFV Architectures

Amina SAHBI[1], Faouzi JAIDI[2], Adel BOUHOULA[3]
University of Carthage, Higher School of Communications of Tunis,
LR11TIC03 Innov'COM Innovation of Communicant and Cooperative Mobiles Laboratory,
LR18TIC01 Digital Security Research Lab, Tunis, Tunisia[1,2]
University of Carthage, National School of Engineers of Carthage, Tunis, Tunisia[2]
Arabian Gulf University, Department of Computing, University of Carthage, Kingdom of Bahrain[3]

***Abstract*—The integration of Software Defined Networking (SDN) and Network Function Virtualization (NFV) offers considerable advantages in terms of scalability, interoperability, and cost-efficiency. They redefine network architecture, replacing rigid hardware-based control with a more flexible, software-driven approach. However, this convergence also introduces significant security threats and challenges due to architectural vulnerabilities and an expanded attack surface. This study presents a comprehensive overview of the key security risks associated with SDN/NFV networks. It analyzes existing countermeasures, highlighting their effectiveness in addressing specific threats while identifying limitations in achieving comprehensive security due to inherent architectural vulnerabilities. The study concludes with a discussion on open challenges and future research directions toward more secure and resilient network infrastructures. This study highlights the importance of an integrated security approach and identifies areas where further research is required to enhance SDN/NFV security.**

***Keywords*—Next generation network security; software defined networking; network function virtualization; network security; artificial intelligence**

I. Introduction

In today's hyper-connected world, networks have become the backbone of modern communications and services, from enterprise operations to personal applications. As networks grow in size and complexity, traditional networking architectures face limitations in terms of scalability, flexibility, and manageability. Therefore, centralized administration and virtualized infrastructures have become essential for effective remote management and configuration [1].

This has led to the development of SDN, a novel paradigm that decouples the control plane from the data plane, offering greater programmability. SDN provides a centralized view of the entire network, enabling administrators to dynamically manage network traffic and resources through software applications. By separating control functions from the physical infrastructure, SDN allows the rapid deployment of new network services, particularly in environments such as data centers, cloud computing, and large-scale enterprise networks [2].

While SDN centralizes and simplifies network control, NFV focuses on decoupling network services from physical devices by running them as software applications on virtualized platforms. Through this change, hardware costs are reduced, scalability is enhanced, and service provision is accelerated. Combining SDN with NFV provides a more responsive and controlled network architecture, especially for dynamic environments [3].

Despite these promising benefits, the integration of SDN/NFV causes new vulnerabilities and security issues. The centralized control in SDN makes controllers potential targets for cyber attacks [4]. Furthermore, open interfaces such as Northbound and Southbound Application Programming Interfaces (APIs) expand the attack surface, creating more entry points for attackers [5].

Given these emerging security challenges, traditional security mechanisms become inadequate. Artificial Intelligence (AI) has increasingly become essential to detect, mitigate, and prevent sophisticated attacks proactively [6]. Similar perspectives have been developed in our prior works, where AI and Machine Learning (ML) were applied to enhance SDN/NFV security [7], [8]. AI techniques enable automated anomaly detection, adaptive threat response, and intelligent decision-making that considerably surpass the capabilities of conventional security solutions.

To frame the scope of this research, our study is guided by a central inquiry: How can intelligent defense mechanisms enhance the security of SDN and NFV architectures by addressing their inherent vulnerabilities and evolving threats? More specifically, we ask in what ways such defenses can strengthen these architectures against critical weaknesses and emerging cyberattacks, and furthermore, how next-generation intelligent strategies may be leveraged to secure SDN and NFV while overcoming their most pressing challenges. These complementary perspectives establish the foundation for the analysis and recommendations presented in this work.

This study contributes by offering a thorough examination of security vulnerabilities within SDN and NFV ecosystems, a critical review of AI-based defense strategies, and the identification of research gaps that hinder robust implementation. Furthermore, it outlines key recommendations to guide future developments toward more resilient and secure programmable networks.

This study intends to encourage further studies towards enhancing the security of next-generation network environments by highlighting the need of an integrated security strategy. In our study, we provide an overview of the main security risks influencing network topologies based on SDN and NFV. To address these vulnerabilities, we review the countermeasures that have been proposed in the literature and assess their effectiveness and limitations. In addition, we discuss current research gaps and open challenges that may hinder the implementation of reliable and secure SDN/NFV networks.

The structure of this study is as follows : Section II introduces background and foundational concepts related to SDN and NFV, followed by an overview of major security threats and challenges in Section III. Section IV presents countermeasures and a discussion on future directions. Section V provides concluding remarks and perspectives for future research.

II. Background and Fundamentals

A. From Traditional Network to Software-Defined Networking

The rapid progress of digital technologies, the virtualization of services, and data intensive applications has revealed the limitations of traditional network infrastructures [9]. Traditional networks rely on a distributed control model where configuration and management tasks, such as routing policies, access control rules, and software updates, must be applied individually to each device. This lack of centralized orchestration raises operational complexity, extends the time required to deploy updates or resolve misconfigurations, and leads to higher maintenance costs [10]. Additionally, these architectures frequently suffer from insufficient use of computational and bandwidth resources, as well as a restricted capacity.

SDN emerged as a solution to overcome these limitations. By separating the control plane from the data plane, SDN enables centralized network control and simplifies the deployment of new services, as illustrated in Fig. 1.

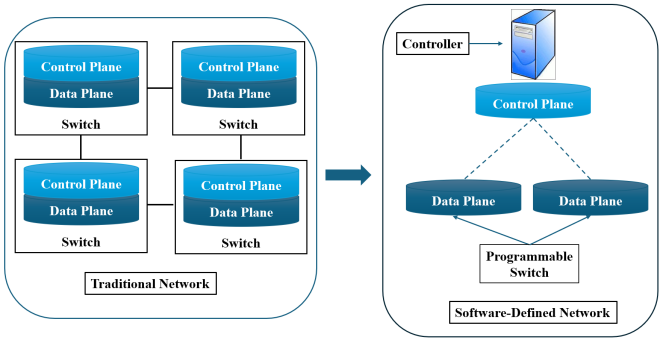


Fig. 1. Traditional Network vs Software-defined networks.

The main benefit of SDN is its ability to decouple network control from hardware, allowing administrators to manage and modify the network through software interfaces without need for physical alterations [11]. The abstraction is improved by the vendor-neutral architecture of SDN, which ensures seamless interoperability across multiple network devices and platforms. This allows enterprises to extend their infrastructure, optimize performance on customer demand, and respond quickly to operational changes or security incidents [11]. The inherent programmability and adaptability of SDN environments also contribute to significantly better resource utilization, allowing networks to operate more efficiently and respond dynamically to real-time demands [12].

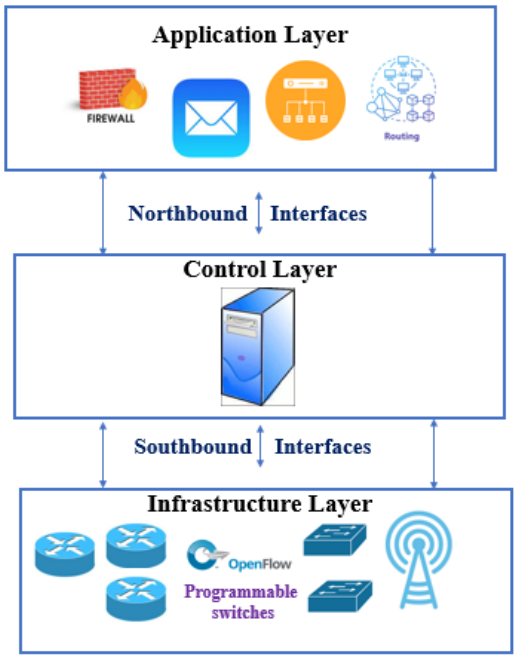


Fig. 2. Software-defined networks architecture.

Fig. 2 displays the fundamental architecture of SDN. It consists of three main layers: the application layer, the control layer, and the infrastructure or the data layer [13]. These layers are connected to one another by well-defined interfaces, known as northbound and southbound interfaces. This ensures that interactions and communication are carried out without any disruptions across the framework. Network behavior and policies are defined by software services hosted at the application layer. These services may include traffic engineering, quality of service (QoS) management, intrusion detection, load balancing, and security monitoring. Instead of explicitly configuring hardware, applications communicate policies to the controller through the northbound interface. This abstraction allows developers to introduce new features or security functions rapidly without being constrained by the limitations of proprietary hardware. In SDN systems, this layer is ultimately responsible for enabling innovation and service agility [14].

The control layer is the most important part of SDN. It makes decisions by transforming application-level requirements into precise routing rules [14]. The SDN controller has a global, real-time view of the whole network, which allows it to make smart decisions about how traffic should be routed or managed. This layer converts high-level application requirements into low-level forwarding rules, which are installed dynamically on switches and routers. By decoupling decision-making from the physical devices, the controller ensures coherent policy enforcement, optimized resource usage, and rapid adaptation to network changes or security events. More-

over, this layer supports programmability, allowing network operators to define behavior through APIs and automation frameworks.

The last layer, known as the infrastructure layer, also referred to as the data plane, consists of physical or virtual forwarding devices, such as switches and routers, that execute the instructions received from the controller through the southbound interface [15]. These devices are simplified to focus exclusively on packet forwarding, which reduces their complexity, cost, and management overhead. The controller may change their behavior dynamically, allowing for precise traffic flow management and effective network resource use.

These layers together represent the four basic principles of SDN: network programmability, abstraction of physical infrastructure, centralized intelligence, and the separation of control and data planes. Supported by protocols such as OpenFlow, this architecture simplifies network management, enhances scalability, and delivers the flexibility needed to respond quickly to evolving operational and security demands.

B. The Need for Network Function Virtualization

Although SDN significantly enhances network control by decoupling the control logic from forwarding devices, it does not virtualize the network services themselves [16]. Middleboxes such as load balancers, firewalls, and NAT devices are still dependent on proprietary hardware. This limits flexibility and drives up operational costs. To address these constraints and fully exploit the paradigm of SDN, NFV was developed.

The goal of NFV is to enable network functions to perform as software instances on common, commodity servers by separating them from specialized hardware [17]. This change removes reliance on specific vendor appliances and allows operators to deploy, scale, and manage services flexibly. SDN offers centralized intelligence, and NFV delivers virtualized services, creating an agile and cost effective networking environment.

According to the European Telecommunications Standards Institute (ETSI), NFV is built upon three essential key components, illustrated in Fig. 3.

The components include Virtual Network Functions (VNFs), Network Function Virtualization Infrastructure (NFVI), and the Management and Orchestration (MANO) system. Together, NFVI, VNFs, and MANO enable a software-driven network that is automated and easily adaptable to operational demands.

The first is the NFVI, which includes the physical hardware, virtualization layer, and the virtualized resources. It serves as the foundational layer, integrating the physical hardware (servers, storage, and network resources) with a virtualization layer that abstracts these resources into pools of compute, storage, and connectivity. This abstraction allows for the dynamic allocation of virtualized resources, supporting the instantiation of virtual machines (VMs) and containers that host the network functions [18]. This layer provides the foundation to host virtual machines and containers.

The second block is VNFs, which implement specific network functions using the virtual resources offered by the

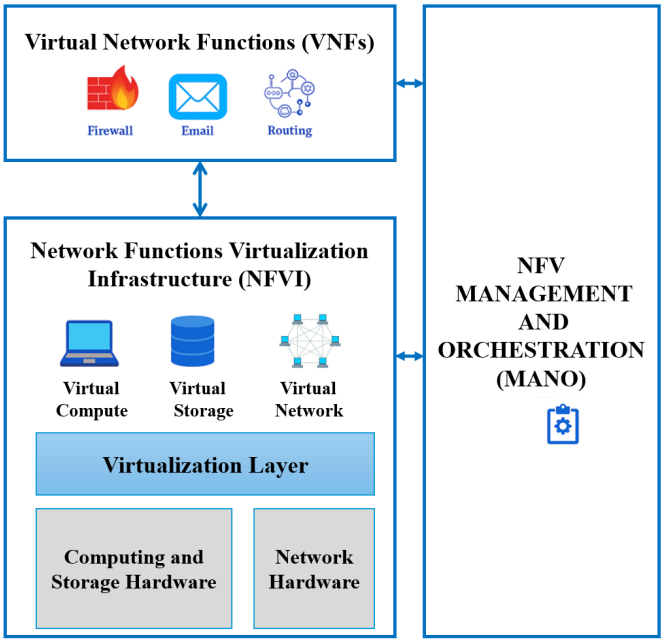


Fig. 3. Network function virtualization.

NFVI. Examples include virtual routers (vRouter), firewalls (vFW), and NAT (vNAT). These VNFs are modular and can be deployed, updated, or removed dynamically [18].

The third and most critical block is MANO [18]. MANO interacts with both the VNFs and the NFVI to manage the lifecycle of services and resources. It is responsible for automating tasks such as deployment, scaling, monitoring, and resource allocation. This block ensures that virtual services can be managed efficiently and adjusted in real-time based on traffic demands or system policies.

NFV offers several benefits, as shown in Fig. 4:

By replacing dedicated appliances with software instances, NFV reduces capital expenditures (CAPEX) and operational expenditures (OPEX) while accelerating service deployment. It also enables faster deployment of new services and better scalability, since virtual functions can be instantiated or terminated as needed. In addition, NFV simplifies updates and promotes vendor independence, allowing service providers to choose flexible software solutions without being locked to specific hardware platforms [19]. These characteristics position NFV as a key enabler of flexible, automated, and future-proof network infrastructures.

C. SDN and NFV Complementarity

SDN and NFV are two complementary technologies that aim to modernize and optimize network architecture. While SDN focuses on the separation of the control and data planes to enable centralized overview and programmable control, NFV addresses the virtualization of network services that traditionally relied on dedicated hardware.

These two paradigms operate at different layers of the network but are designed to work together [20]. SDN provides the intelligence and global view necessary to control

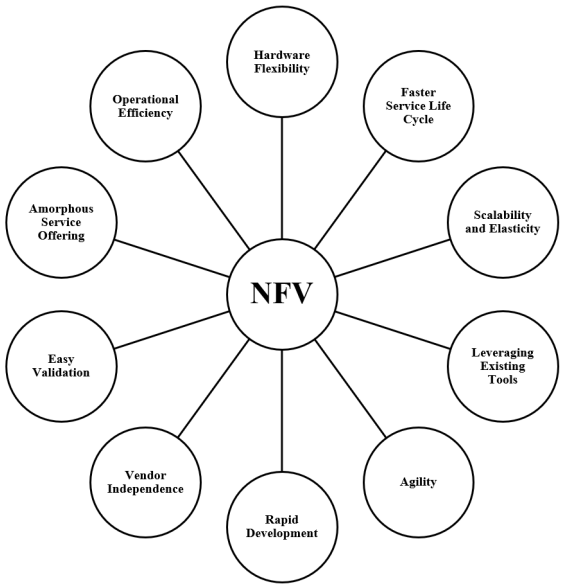


Fig. 4. Advantages of network function virtualization.

how data flows across the network, while NFV delivers the actual network functions as software running on virtualized infrastructure. The integration of SDN and NFV results in a flexible and programmable networking model, where services and control are both decoupled from rigid infrastructures and can evolve dynamically to meet operational requirements.

This synergy allows for greater agility, faster service deployment, and reduced operational complexity. It also supports automation and resource optimization by enabling flexible service chaining and centralized orchestration. The combination of SDN and NFV is particularly beneficial in modern use cases such as 5G, cloud-native infrastructure, and edge computing, where scalability and adaptability are essential [21]. To better understand the roles and differences of SDN and NFV, Table I presents a comparative summary of their key aspects.

TABLE I. COMPARISON BETWEEN SDN AND NFV

Aspect	SDN	NFV
Primary Focus	Separation of control and data planes	Virtualization of network functions
Key Component	SDN Controller	VNFs, NFVI, MANO
Control Mechanism	Centralized control via software	Lifecycle and resource management via MANO
Infrastructure	Programmable switches and routers	Servers and virtualization layers
Deployment Speed	Faster path control and policy updates	Rapid deployment of virtual services
Vendor Dependency	Vendor-neutral (via OpenFlow and APIs)	Reduces reliance on proprietary hardware
Example Use Cases	Dynamic routing, traffic engineering	Virtual firewalls, NAT, load balancing
Relationship	Controls how data flows in the network	Provides the services that flow over the network infrastructure

While SDN enables centralized control and dynamic network management, it does not cover the virtualization of the network functions themselves. For this reason, a complementary approach called NFV was introduced to decouple network services from dedicated hardware. NFV enables these services such as routing, firewalls, or load balancing to run as software on general purpose servers.

III. SECURITY THREATS AND CHALLENGES

A. Software-Defined Networking Challenges

SDN introduces a revolutionary network architecture; however, this change brings several challenges. One of the most critical is the centralization of control. Acting as the brain of the network, the controller becomes a single point of failure. Any malfunction, misconfiguration, or successful intrusion against this component can lead to a complete collapse of network services. Attackers who manage to compromise the controller gain unprecedented control, enabling them to alter flow rules, inject malicious policies, and disrupt or hijack traffic across the entire infrastructure. If compromised, the controller could enable full network disruption [22]. In this context, our earlier research proposed intelligent approaches to detect and resolve policy violations within SDN control and data planes [23], [24].

In addition, communication between SDN components introduces another layer of vulnerability. The southbound interface, typically relying on protocols such as OpenFlow [25], lacks robust native security features, leaving it exposed to message spoofing, tampering, and man-in-the-middle attacks (MitM). Similarly, northbound interfaces often suffer from insufficient access control. When improperly secured, these interfaces may allow unauthorized access or the injection of malicious services which significantly weakens the trust model of the entire architecture.

Scalability [26], also represents a persistent challenge. As networks grow in size and complexity, optimizing controller placement, balancing workloads, and minimizing latency in control messages become increasingly challenging. In distributed deployments, where multiple controllers must synchronize to maintain a coherent global view, inconsistencies may arise, resulting in policy conflicts and operational instability [27]. These architectural weaknesses are aggravated by the absence of unified security frameworks and the reliance on open-source components that often lack rigorous hardening against advanced threats.

A particularly severe threat to SDN infrastructures is the Distributed Denial of Service (DDoS) attack. In traditional networks, such attacks aim to overwhelm servers or network links with excessive traffic, but in SDN, the impact is often more damaging because of the centralized nature of control. During a DDoS attack, large volumes of malicious traffic are generated from multiple sources and directed toward the data plane, triggering a flood of control messages sent to the controller. This sudden surge of requests can exhaust the controller's processing capacity, resulting in delays, instability, or even total network failure. Attackers may also target the controller directly, exploiting its role as the decision-making hub to paralyze its operations. The consequences of a successful DDoS attack include severe service degradation, unavailability of critical applications, and disruption of communication across the entire network [28].

Beyond these aspects, SDN operates in a highly dynamic environment where policies, flows, and states change continuously. Ensuring consistent visibility across all elements, preventing rule conflicts, and maintaining traceability in such rapidly evolving conditions require sophisticated monitoring systems that are still under development [29]. Without adequate oversight, misconfigurations or silent intrusions can remain undetected for extended periods, undermining the overall security posture of the network.

Table II summarizes the main threats affecting each SDN domain, outlines the associated risks, and highlights defensive measures capable of mitigating these vulnerabilities. This analysis emphasizes the urgent need for integrated security mechanisms capable of addressing both the architectural weaknesses and the evolving threat landscape that characterizes SDN deployments [30].

B. Network Function Virtualization Challenges

NFV reshapes traditional network architectures by decoupling network functions from proprietary hardware and deploying them as software instances on commodity servers [19]. This transformation provides unprecedented scalability and operational flexibility, but at the same time introduces complex risks that threaten the reliability and security of virtualized infrastructures [31]. The dynamic and distributed nature of NFV, where VNFs are instantiated, migrated, or terminated on demand, creates an environment in which ensuring consistency, isolation, and performance becomes highly challenging. Unlike static hardware-based systems, NFV relies on virtualization layers, orchestration components, and multi-domain coordination, all of which may become sources of vulnerabilities if not carefully secured [31].

The threats associated with NFV extend across multiple layers, from the physical hardware to virtualized resources and orchestration control, each requiring tailored countermeasures. Table III consolidates the main threat categories, describes their mechanisms, and links them to appropriate mitigation strategies. This table demonstrates that achieving security in NFV is not the responsibility of a single component but rather the result of coordinated efforts across all layers, combining isolation, authentication, monitoring, and resilience techniques to ensure that flexibility does not come at the expense of security.

One major concern is the complexity of orchestration and management of virtualized resources across multiple domains. The dynamic nature of VNFs makes it difficult to ensure consistent performance, availability, and isolation. Furthermore, resource allocation and scaling mechanisms can lead to unpredictable performance and increased attack surfaces. Another challenge lies in interoperability and standardization. NFV environments often consist of heterogeneous components from different vendors, making integration and management more error prone. Additionally, the reliability and fault tolerance of VNFs must be guaranteed, especially in critical applications such as vehicular or industrial networks.

The orchestration of VNFs across heterogeneous infrastructures is one of the primary difficulties in NFV deployment. Managing resources dynamically while maintaining service continuity requires precise coordination of computing, storage, and network elements, a task that becomes increasingly complex in multi-tenant or cross domain environments. Inconsistent resource allocation, poor synchronization, or errors during VNF migration may lead to unpredictable QoS, degradation in performance, and in some cases complete service disruption [32]. Furthermore, the lack of standardized interfaces and uniform protocols between components from different vendors hampers interoperability, making integration error prone and potentially insecure. These architectural constraints are even more critical in latency sensitive environments such as industrial control systems and vehicular networks, where reliability and fault tolerance are non negotiable requirements [33].

From a security perspective, NFV inherits threats from both traditional networks and cloud computing while introducing new attack surfaces specific to virtualization. Hypervisors, which manage the virtualization layer, are particularly sensitive targets. Also, a successful exploitation can compromise all hosted VNFs and give attackers control over the underlying infrastructure. The orchestration layer, known as MANO, also becomes a high-value target because of its privileged role in managing the entire NFV lifecycle. Exploiting insecure APIs or misconfigurations at this level can lead to unauthorized operations, data leaks, or the injection of malicious components [34]. The trustworthiness of third-party VNFs is another major concern, as these functions may contain backdoors, hidden malware, or exploitable vulnerabilities that compromise the entire chain of services once deployed.

Another significant challenge arises from the lifecycle of VNFs. Each instantiation, scaling, migration, or termination presents specific vulnerabilities [35]. For example, during migration, data and states are transferred across domains, and insufficient protection during this process can allow interception or tampering. Similarly, insecure onboarding of VNFs can facilitate the deployment of malicious images, while improper termination may leave residual data or open configurations exploitable by attackers. The absence of continuous monitoring during these operations further increases the likelihood of undetected intrusions or misconfigurations persisting within the infrastructure [36].

The complexity of NFV also lies in its dependency on cloud-like environments and programmable interfaces, where the attack surface expands rapidly with each added component or tenant. An adversary can exploit poorly monitored orchestration decisions, manipulate inter-VNF communications, or overload physical resources to trigger cascading failures. These scenarios highlight the urgent need for context-aware intrusion detection, runtime verification mechanisms, and automated incident response capable of adapting to evolving threats. At the same time, strict verification of VNF integrity, secure software supply chains, and robust cryptographic protections for management channels are essential to strengthen trust within NFV ecosystems. Fig. 5 highlights some of the most critical threats affecting NFV environments. These threats arise at multiple layers of the architecture, ranging from physical resources to virtualization platforms and orchestration mechanisms, underscoring the need for comprehensive and coordinated defense strategies.

TABLE II. RISKS AND THREATS IN SDN SECURITY DOMAINS

Security Component	Vulnerability/Threat	Risk Consequences	Defensive Action
Control Plane	Over dependence on a single controller instance	Loss of orchestration, network paralysis	Use distributed architectures, backup controllers, and high availability frameworks
Northbound Interfaces	APIs exposed to orchestration platforms and apps without strict security controls	Unauthorized access, policy override, service injection	TLS encryption, API gateway, strong RBAC, input validation
Southbound Interfaces	Message injection or tampering in protocols like OpenFlow	Corrupted flow tables, misrouting, loss of trust	Message authentication, channel encryption, switch-level access policies
Application Layer	Malicious or vulnerable apps interacting with the controller	Flow conflicts, data leakage, service disruption	App sandboxing, behavioral monitoring, application certification workflows
Flow Management	Conflicts or outdated flows in switch tables	Policy violation, packet loops, resource waste	Implement rule expiration, rule audits, consistency validation engines
Inter Communication	Lack of verification between East and West control domains	Propagation of compromised state, unsynchronized behavior	Signed messaging, secure sync protocols, consensus mechanisms
Monitoring	Unlogged configuration changes, missing traceability	Persistent misconfigurations, undetected policy violations	Continuous auditing, automated logging, version control and rollback systems

TABLE III. VIRTUAL NETWORK SECURITY: THREATS VS COUNTERMEASURES

Threat Category	Threats	Countermeasure Category	Countermeasures
Disclosure	Information Leakage, Information Interception, Introspection Exploitation	Access Control	Trusted Virtual Domains, Sandboxes
Deception	Identity Fraud, Loss of Registry Entries, Replay Attacks	Authentication	Interoperability Between Federations, Certificate-Based, Key-Based
Disruption	Physical Resource Overloading, Physical Resource Failure	Confidentiality	VLANs and VPNs, Tunneling and Cryptography, Firewalling and Subnetting
Usurpation	Identity Fraud, Software Vulnerability Exploitation	Integrity, Nonrepudiation, Availability	Path Splitting, Limiting Introspection, Cryptography, Timestamping, Physical Resource Isolation, Virtual Network Resilience

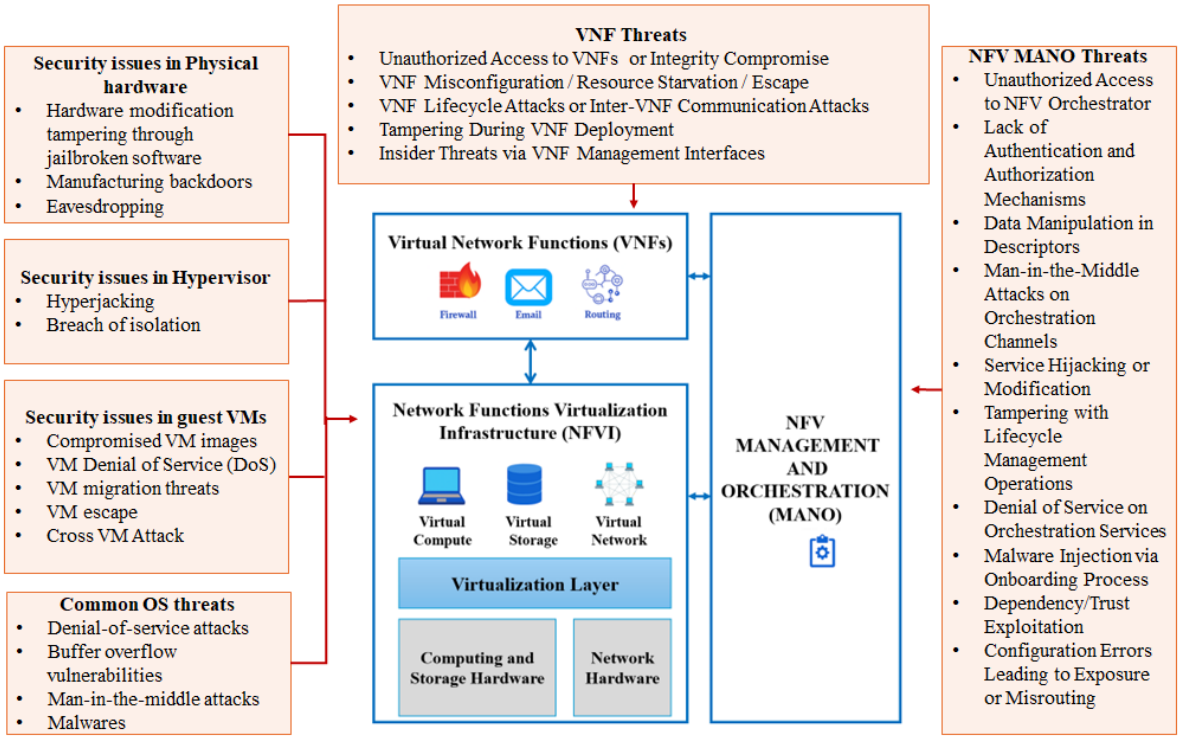


Fig. 5. Network function virtualization threats.

C. Security Threats within SDN and NFV

Nowadays, the majority of existing frameworks are focused on migrating network functions from hardware devices to virtualization environments. SDN and NFV are two independent but complementary concepts. When combined, they provide an additional benefit in terms of simplified management, rapid service deployment, and reduced operational costs. However,

it also significantly expands the attack surface, as each technology introduces its vulnerabilities while their interaction creates new, composite risks that span multiple layers of the architecture [36].

In SDN, the centralized controller is an essential asset but also a critical weakness. Its compromise can disrupt the entire network, enabling attackers to manipulate flow rules or shut down services altogether. Similarly, NFV introduces threats at the virtualization layer, where hypervisors and VNFs become primary targets. Malicious VNFs or hypervisor exploits may breach isolation boundaries, gaining access to the host environment and other tenants. The orchestration layer, shared between SDN and NFV, faces its own set of threats. Weakly secured MANO components or misconfigured APIs can allow privilege escalation, unauthorized control operations, or the deployment of rogue VNFs that compromise the network.

MitM attacks [37], represent another severe risk in these ecosystems. Interception of traffic can occur both in SDN control channels and during NFV operations such as VNF migration, allowing adversaries to manipulate or spy on communications. DoS attacks are also particularly damaging in this context. Overwhelming the SDN controller, the NFV orchestrator, or the underlying infrastructure can degrade performance or even cause complete service outages. These attacks become more difficult to mitigate in dynamic environments, where large volumes of traffic are legitimate and distinguishing malicious flows is challenging.

The programmability of SDN and NFV, while central to their flexibility, also creates opportunities for misconfiguration. Incorrect policies, insecure automation scripts, and flawed orchestration workflows can unintentionally expose the network to vulnerabilities without any direct attacker involvement. Furthermore, unpatched software in SDN applications or VNF packages provides attackers with exploitable weaknesses that can propagate rapidly across virtualized infrastructures. Third-party VNFs pose an additional risk, as they may include hidden backdoors or malicious code capable of undermining the integrity of the network once deployed [38].

The integration of SDN and NFV introduces complex interdependencies between control, orchestration, and virtualization layers. This tight coupling means that an attack on one component can cascade, affecting other layers and amplifying its impact. Securing such an environment requires more isolated defense mechanisms. It demands a holistic approach, where isolation, strong authentication, integrity verification, and continuous monitoring are enforced consistently across all layers. Real-time intrusion detection, secure orchestration policies, and rigorous VNF verification processes are crucial for ensuring that the benefits of SDN/NFV adoption are not undermined by evolving cyber threats [39].

Table IV synthesizes the main threat domains observed in SDN and NFV environments, and highlights how their integration increases exposure to attacks. This summary underlines the necessity for coordinated security mechanisms capable of addressing vulnerabilities at every level of the SDN/NFV stack, preventing isolated weaknesses from escalating into systemic failures.

TABLE IV. SUMMARY OF SECURITY CHALLENGES IN SDN AND NFV

Domain	Challenges / Threats
NFV	VNF isolation failure, insecure lifecycle management, hypervisor vulnerabilities, untrusted VNF onboarding
SDN	Centralized controller exposure, insecure APIs, third-party app vulnerabilities, southbound protocol threats
SDN/NFV Integration	DoS attacks, MitM, VNF escape, orchestration manipulation, privilege escalation

IV. COUNTERMEASURES REVIEW AND FUTURE DIRECTIONS

A. Countermeasures Review

In recent years, research on SDN and NFV security has evolved considerably, moving from traditional protection mechanisms to intelligent, adaptive frameworks. Countermeasures have been proposed for every architectural layer, including the control plane, data plane, virtualization infrastructure, orchestration systems, and exposed APIs.

For SDN, earlier studies recommended strengthening the controller through redundancy, fine-grained access controls, and secure southbound communication. Jiménez et al. [40] applied the STRIDE methodology to analyze threats across SDN layers and recommended tighter access policies, controller hardening, and secure APIs. Despite these enhancements, they noted unresolved challenges such as dynamic policy conflicts and insufficient protection against zero-day vulnerabilities.

Beyond static defenses, several studies have proposed AI-enhanced security frameworks. Shobowale et al. [41] explored 5G threats focusing on SDN and NFV layers, introducing a combination of blockchain-based authentication, layered access control, and AI-driven intrusion detection. However, their framework faced scalability challenges, high computational overhead, and limited coverage against emerging side-channel and supply-chain attacks.

In the NFV context, Pattaranantakul et al. [42] proposed countermeasures such as hypervisor isolation, trust domains, and adaptive access control. While these techniques improved security at individual layers, the lack of orchestration-level integration allowed sophisticated interlayer attacks to persist.

Other research explored entropy-based detection. Fan et al. [43] developed a lightweight approach using fusion entropy to detect DDoS attacks in SDN. Their model combined information entropy and logarithmic energy entropy to identify anomalies with high precision. Although effective in simulations, it required manual threshold settings and lacked evaluation in complex multi-controller deployments.

More recent works emphasized hybrid and intelligent systems. Nadeem et al. [44], reviewed ML techniques for securing SDN, focusing on detecting DDoS, botnets, and ransomware. They demonstrated how supervised and deep learning models can improve detection accuracy, while also highlighting vulnerabilities to adversarial samples and the need for explainability. Similarly, the Hybrid SDN-IDS framework leveraged dynamic SDN control and NFV monitoring to detect and mitigate threats in real-time. Its evaluation showed improved detection rates, but its performance under large-scale traffic conditions remains uncertain.

TABLE V. FUTURE RESEARCH DIRECTIONS FOR ENHANCING SDN/NFV SECURITY

Research Direction	Objective	Addressed Challenge
Unified Threat Intelligence [40], [46]	Enable cross-layer visibility and event correlation across SDN, NFV, and cloud domains	Fragmented monitoring, lack of coordination between components
Lightweight, Real-Time Detection [43], [44]	Design efficient anomaly detection models suitable for resource-constrained environments	High computational cost of current AI methods; unsuitable for IoT/edge
Autonomous and Self-Adaptive Security [47]	Develop self-healing systems that react to evolving threats and reconfigure policies automatically	Inability to respond dynamically to zero-day or stealth attacks
Secure VNF Marketplaces [19], [42]	Ensure trust in third-party VNFs through attestation, certification, and runtime monitoring	VNF tampering, backdoors, and insecure onboarding
Formal Verification of Orchestration Policies [41]	Apply formal methods to validate orchestration rules before deployment	Misconfiguration risks, policy inconsistencies, privilege escalation
Post-Quantum Cryptography and Trusted Execution Environments (TEEs) [48]	Strengthen data confidentiality and control-plane integrity in next-gen programmable networks	Future cryptographic attacks; lack of hardware-assisted security primitives

For Next-Generation Networks, AI-driven cross-layer frameworks have gained traction. Alnaim [45] critically examined SDN, NFV, and network slicing in 5G, proposing security-by-design methodologies and UML-based reference models. However, the lack of standardized enforcement limits their adoption. The study [46] proposed integrating SDN controllers, NFV orchestration, and AI anomaly detection for slice-aware protection in 5G/6G. While promising, its real-world latency and scalability remain to be validated. Complementary to these studies, our contributions investigated how ML-based algorithms can strengthen intrusion detection in SDN/NFV environments [8] and proposed frameworks for automated policy enforcement [23].

Finally, Li et al. [47] introduced a proactive defense combining Moving Target Defense (MTD) and Reinforcement Learning (RL). This approach dynamically reconfigures network parameters to counter stealthy attacks, yet introduces new risks tied to the vulnerabilities of RL models themselves.

Although several AI-based security mechanisms have been proposed in the literature, most are evaluated in ideal or simulated settings. There remains a lack of critical assessment under adversarial scenarios or across heterogeneous infrastructures combining SDN controllers, NFV orchestrators, and multi-tenant environments.

In summary, literature shows a clear progression toward adaptive, AI-enhanced, and cross-layer security frameworks for SDN and NFV. However, common limitations persist such as high computational demands, lack of standardized orchestration, and limited validation under real world conditions. These findings highlight the need for lightweight AI models, resilient orchestration-level controls, and robust defenses against adversarial threats.

B. Discussion and Future Directions

The reviewed literature confirms a clear evolution toward AI-augmented, cross-layer, and adaptive security frameworks for SDN and NFV networks. These solutions aim to address complex threats in highly dynamic environments such as 5G, IoT, and vehicular networks. Nevertheless, several persistent challenges remain unresolved.

Among AI techniques, supervised models dominate current literature, yet their dependence on labeled data limits its real-world generalization [44]. RL and unsupervised approaches, while promising, still lack extensive benchmarking in SDN/NFV contexts [47]. This observation is consistent with earlier surveys that emphasized the gap between simulation-based validation and practical deployment in operational infrastructures [40], [44]. For instance, [40] highlights how current solutions often remain limited to laboratory environments, reducing their applicability to real-world programmable infrastructures.

Another key limitation lies in the absence of standardized and interoperable security frameworks. Current solutions often address threats at isolated layers (control, data, or virtualization), but fail to ensure consistent protection across the entire architecture [41], [46]. Moreover, many approaches are validated only under ideal laboratory conditions, with limited assessment under realistic network traffic and latency scenarios [45]. This limitation is particularly evident in large-scale SDN deployments, where orchestration complexity and latency constraints cannot be ignored [46].

AI and ML have shown significant potential in improving detection and response capabilities, as demonstrated in recent works on SDN security [44], [45]. However, they bring new risks such as high computational demands, lack of explainability, and exposure to adversarial manipulation [25], [47]. The complexity of training and deploying robust AI models remains a barrier, particularly in edge and multi-tenant scenarios where resources are constrained, a limitation which is also emphasized in studies on real-time SDN/NFV security frameworks [41], [46]. These findings are consistent with our previous proposals, which stressed the importance of lightweight and explainable AI for practical deployment in programmable networks [7], [24]. As noted in [47], adversarial interference against AI-driven systems represents an emerging risk that requires more resilient and transparent defense mechanisms.

Moreover, a critical gap remains in evaluating these security frameworks under realistic, heterogeneous environments. Very few approaches have been tested across operational SDN/NFV infrastructures involving cross-domain orchestration, adversarial interference, or third-party VNF integrations factors, that are crucial for assessing their robustness and scalability.

In summary, while substantial progress has been made toward intelligent and adaptive SDN/NFV defenses, the lit-

erature highlights clear research gaps. Addressing interoperability, ensuring scalability, and developing robust, adversarial-resilient AI models remain essential for advancing secure and trustworthy programmable networks.

To guide future work, several promising directions have emerged from the literature. These focuses on enhancing security intelligence, improving the adaptability of defenses, and addressing the unique challenges of highly dynamic SDN/NFV ecosystems. Table V summarizes these key directions and their corresponding objectives.

In addition to these conceptual directions, future work will include experimental validation of selected approaches, particularly AI-driven detection mechanisms, through implementation and evaluation in real-world programmable network environments such as SDN-based vehicular or cloud-edge architectures.

In conclusion, while existing studies have laid a strong foundation, the increasing complexity of SDN and NFV demands new generations of security frameworks that are not only intelligent and adaptive but also lightweight, verifiable, and resilient to emerging attack strategies.

V. Conclusion

Next-Generation Networks, built upon concepts like SDN and NFV, have emerged as innovative and diverse architectures. By decoupling the control and data planes and virtualizing network functions, they offer unprecedented flexibility, programmability, and operational efficiency. However, this paradigm shift also brings forth new attack vectors, architectural weaknesses, and orchestration challenges that demand proactive and adaptive security measures.

This study underscores the necessity of adopting a unified and cross-layer security framework to mitigate the evolving threats specific to SDN and NFV deployments. By identifying critical gaps in existing solutions and evaluating their limitations, it aims to inspire more effective approaches tailored to the demands of programmable and distributed infrastructures. Furthermore, the integration of intelligent techniques, such as AI and ML, emerges as a promising direction offering dynamic, context-aware defenses suited for complex environments such as 5G, IoT, and edge networks.

In line with our guiding research questions, this work has shown that intelligent defense mechanisms can enhance the security of SDN and NFV by addressing their inherent vulnerabilities. They can also strengthen these architectures against emerging cyberattacks, while next-generation intelligent strategies hold strong potential to overcome their most pressing challenges. Nevertheless, certain limitations remain: most proposed solutions are evaluated in simulated settings, lack cross-layer interoperability, and do not fully consider adversarial manipulation or real-world scalability.

Future research must therefore prioritize resilient, lightweight, and verifiable security mechanisms validated under realistic deployment conditions. By closing these gaps, the community can move closer to achieving robust and trustworthy programmable networks capable of supporting critical infrastructures in 5G, IoT, and beyond.

References

[1] Awais, Muhammad, et al. "Comparative analysis of traditional and software defined networks." 2021 Mohammad Ali Jinnah University International Conference on Computing (MAJICC). IEEE, 2021.

[2] Çetin, Ahmet, Derya Gültekin, and Nihan Yıldırım. "Implications of NFV-SDN technology: An exploratory study of Turkish telecom industry." Journal of Global Information Technology Management 28.2 (2025): 111-135.

[3] Herrera, Juliver Gil, and Juan Felipe Botero. "Resource allocation in NFV: A comprehensive survey." IEEE Transactions on Network and Service Management 13.3 (2016): 518-532.

[4] Ebadinezhad, Sahar, and Pierre Fabrice Nlend Bayemi. "SDN and NFV security challenges and solutions for minimizing failures in IoT networks: Literature review." Research Advances in Network Technologies (2025): 22-40.

[5] Xu, Chunxue. "Resource Optimization Algorithm for 5G Core Network Integrating NFV and SDN Technologies." International Journal of Intelligent Networks (2025).

[6] Hatim, J. A. A. D. O. U. N. I., C. H. A. O. U. I. Habiba, and S. A. A. D. I. Chaimae. "Evolving Security for 6G: Integrating Software-Defined Networking and Network Function Virtualization into Next-Generation Architectures." International Journal of Advanced Computer Science & Applications 15.6 (2024).

[7] A. Sahbi, F. Jaidi, and A. Bouhoula, "Artificial Intelligence for SDN Security: Analysis, Challenges and Approach Proposal," in 2022 15th International Conference on Security of Information and Networks (SIN), IEEE, 2022.

[8] A. Sahbi, F. Jaidi, and A. Bouhoula, "Machine Learning Algorithms for Enhancing Intrusion Detection Within SDN/NFV," in 2023 International Wireless Communications and Mobile Computing (IWCMC), IEEE, 2023

[9] Priyadarsini, Madhukrishna, and Padmalochan Bera. "Software defined networking architecture, traffic management, security, and placement: A survey." Computer Networks 192 (2021): 108047.

[10] Abdelrahman, Abdallah Mustafa, et al. "Software-defined networking security for private data center networks and clouds: vulnerabilities, attacks, countermeasures, and solutions." International Journal of Communication Systems 34.4 (2021): e4706.

[11] Domínguez-Dorado, Manuel, et al. "Detection and mitigation of security threats using virtualized network functions in software-defined networks." Applied Sciences 14.1 (2023): 374.

[12] Abd-Allah, Ahmed Gaber Abu, et al. "A Comprehensive Survey on Security Challenges and Solutions in Software-Defined Network." JOURNAL OF COMPUTER SCIENCE (ISSN NO: 1549-3636) 18.04 (2025).

[13] Kenner, Andrea, and Oluwaseyi Oladele. "Software-defined networking (sdn) and network function virtualization (nfv) in telecommunication: A computer science perspective." (2024).

[14] Ashodia, Namita, and Kishan Makadiya. "Detection and mitigation of ddos attack in software defined networking: A survey." 2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS). IEEE, 2022.

[15] Ahmad, Shakir Salman, and Mohammed I. Habelamateen. "Application of Artificial Intelligence and Machine Learning in Software Defined Networks." Journal of Smart Internet of Things 2023.1 (2023): 14-22.

[16] Hasneen, Jehan, and Kazi Masum Sadique. "A survey on 5G architecture and security scopes in SDN and NFV." Applied Information Processing Systems: Proceedings of ICCET 2021. Singapore: Springer Singapore, 2021. 447-460.

[17] Rauf, Bilal, et al. "Application threats to exploit northbound interface vulnerabilities in software defined networks." ACM Computing Surveys (CSUR) 54.6 (2021): 1-36.

[18] Melkov, Dmitrij, and Sarunas Paulikas. "Security benefits and drawbacks of software-defined networking." 2021 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream). IEEE, 2021.

[19] Zahran, Bilal, et al. "Security and Privacy Issues in Network Function Virtualization: A Review from Architectural Perspective." International Journal of Advanced Computer Science & Applications 15.6 (2024).

[20] Razvan, Florea, and Craus Mitica. "Enhancing network security through integration of game theory in software-defined networking framework." International Journal of Information Security 24.3 (2025): 100.

[21] Babbar, Himanshi, et al. "Role of network slicing in software defined networking for 5G: Use cases and future directions." IEEE Wireless Communications 29.1 (2022): 112-118.

[22] Anand, Nemalikanti, et al. "Securing software defined networks: A comprehensive analysis of approaches, applications, and future strategies against DoS attacks." IEEE Access (2024).

[23] Sahbi, Amina, Faouzi Jaidi, and Adel Bouhoula. "Towards a reliable and smart approach for detecting and resolving security violations within SDWN." 2023 International Wireless Communications and Mobile Computing (IWCMC). IEEE, 2023.

[24] Sahbi, Amina, Faouzi Jaidi, and Adel Bouhoula. "An intelligent solution to detect security policy violations in sdn data plane." SYMBOLIC COMPUTATION IN SOFTWARE SCIENCE (2021): 46.

[25] Taheri, Roya, Habib Ahmed, and Engin Arslan. "Deep learning for the security of software-defined networks: a review." Cluster Computing 26.5 (2023): 3089-3112.

[26] Aziz, Israa T., and Ihsan H. Abdulqadder. "An overview on SDN and NFV security orchestration in cloud network environment." Cihan University-Erbil Scientific Journal 5.1 (2021): 20-27.

[27] Pliekhova, Ganna, et al. "Software-configured network architecture vulnerabilities." AIP Conference Proceedings. Vol. 3238. No. 1. AIP Publishing LLC, 2025.

[28] Scott-Hayward, Sandra. "Security-focused Networks of the Future." Proceedings of the 2021 ACM International Workshop on Software Defined Networks & Network Function Virtualization Security. 2021.

[29] Varma, Koppada Durgaprasad, et al. "Software-defined network and network function virtualization: a comparative study." Software-Defined Networking for Future Internet Technology. Apple Academic Press, 2021. 255-276.

[30] Gaur, Kuntal, et al. "Software defined networking: a review on architecture, security and applications." IOP Conference Series: Materials Science and Engineering. Vol. 1099. No. 1. IOP Publishing, 2021.

[31] Mustafa, Zaid, et al. "Intrusion detection systems for software-defined networks: a comprehensive study on machine learning-based techniques." Cluster Computing 27.7 (2024): 9635-9661.

[32] Bilal, Noura, Shavan Askar, and Karwan Muheden. "Challenges and Outcomes of Combining Machine Learning with Software-Defined Networking for Network Security and management Purpose: A Review." The Indonesian Journal of Computer Science 13.2 (2024).

[33] Aditya, T., et al. "Nfv and sdn: A new era of network agility and flexibility." Int. J. Adv. Res. Sci. Commun. Technol (2023): 482-493.

[34] Sanghavi, Preet, Sheel Sanghvi, and Ramchandra S. Mangrulkar. "Software-defined networks a brief overview and survey of services." Software Defined Networking for Ad Hoc Networks (2022): 1-31.

[35] Snehi, Manish, and Abhinav Bhandari. "An SDN/NFV based intelligent fog architecture for DDoS defense in cyber physical systems." 2021 10th International Conference on System Modeling & Advancement in Research Trends (SMART). IEEE, 2021.

[36] Soylu, Mustafa, et al. "NFV-Guard: Mitigating flow table-overflow attacks in SDN using NFV." 2021 IEEE 7th International Conference on Network Softwarization (NetSoft). IEEE, 2021.

[37] Rankothge, W. H., et al. "Network traffic prediction for a software defined network based virtualized network functions platform." 2021 6th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE). Vol. 6. IEEE, 2021.

[38] Piakaray, Divya, et al. "A survey on the utilization of artificial intelligence and machine learning in the field of network functions virtualization and software defined networking." 2023 3rd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE). IEEE, 2023.

[39] Buzhin, I., et al. "Integrity, Resilience and Security of 5G Transport Networks Based on SDN/NFV Technologies." International Conference on Distributed Computer and Communication Networks. Cham: Springer International Publishing, 2021.

[40] Jimenez, Maria B., et al. "A survey of the main security issues and solutions for the SDN architecture." Ieee Access 9 (2021): 122016-122038.

[41] Shobowale, K. O., et al. "Latest advances on security architecture for 5G technology and services." International Journal of Software Engineering and Computer Systems 9.1 (2023): 27-38.

[42] Pattaranantakul, Montida, Chalee Vorakulpipat, and Takeshi Takahashi. "Service function chaining security survey: Addressing security challenges and threats." Computer Networks 221 (2023): 109484.

[43] Fan, Cong, et al. "Detection of DDoS attacks in software defined networking using entropy." Applied Sciences 12.1 (2021): 370.

[44] Nadeem, Muhammad Waqas, et al. "Toward Secure Software-Defined Networks Using Machine Learning: A Review, Research Challenges, and Future Directions." Computer Systems Science & Engineering 47.2 (2023).

[45] Alnaim, Abdulrahman K. "Securing 5G virtual networks: A critical analysis of SDN, NFV, and network slicing security." International Journal of Information Security (2024): 1-21.

[46] Allaw, Zeina, Ola Zein, and Abdel-Mehsen Ahmad. "Cross-Layer Security for 5G/6G Network Slices: An SDN, NFV, and AI-Based Hybrid Framework." Sensors 25.11 (2025): 3335.

[47] Heidarikohol, Niloofar, Shuvalaxmi Dass, and Akbar Siami Namin. "Evolutionary Defense: Advancing Moving Target Strategies with Bio-Inspired Reinforcement Learning to Secure Misconfigured Software Applications." arXiv preprint arXiv:2504.09465 (2025).

[48] García, Carlos Rubio, et al. "Enhancing the security of software defined networks via quantum key distribution and post-quantum cryptography." International Symposium on Distributed Computing and Artificial Intelligence. Cham: Springer Nature Switzerland, 2023.

Bridging Tradition and Technology: Leveraging ERP Systems for Streamlined Supply Chains and Modernized Keropok Lekor Production Management

Faizah Aplop[1], Wan Muhammad Ikhwan Wan Mohammad[2], Muhammad Nasyrul Adly Mohd Afendy[3], Mustafa Man[4], Fakhrul Adli Mohd Zaki[5], Rosaida Rosly[6], Ismail Abu Bakar[7]
Institute of Oceanography and Environment, Universiti Malaysia Terengganu, Kuala Nerus, Terengganu, Malaysia[1,3]
College of Computing, Informatics and Mathematics, UiTM, Kuala Terengganu, Terengganu, Malaysia[2]
Faculty of Computer Science and Mathematics, Universiti Malaysia Terengganu, Kuala Nerus, Terengganu, Malaysia[4,5,6]
ROMA Food Industry Sdn. Bhd., Kuala Nerus, Terengganu, Malaysia[7]

***Abstract*—Online marketplace and social media offers substantial opportunities for business growth for many and it has contributed greatly to the increase demand of keropok lekor (fish cracker) from Terengganu throughout Malaysia as the market reach is increasing. Positive effects of these online platforms as significant digital marketing tools encourage keropok lekor producers in Terengganu to invent and diversify their products with the goal to market and increase the sale of keropok lekor in a larger scale. Some of the innovations are selling keropok lekor in pre-packaged form and introduce different versions of keropok lekor by adding more flavors, textures and shapes to meet broader range of customer preferences. This positive development promotes the commercialization of keropok lekor which then requires the producers like ROMA Food Industry Sdn. Bhd. (RFI) to handle higher market demand without significant disruptions. An automated approach is crucial for them to streamline the keropok lekor business operations to enable them handle not only the increased product market demand, but at the same time the volumes of work or expansion without compromising quality or efficiency. ROMAns is an Enterprise Resource Planning (ERP) system built to optimize keropok lekor business processes by facilitating the flow of information across different functions, improve efficiency, and gain a competitive edge by leveraging integrated data and streamlined operations across keropok lekor business operations.**

***Keywords*—Overall equipment effectiveness; real-time information; centralization; data-driven workflows; business digitalization**

I. Introduction

Originated from Terengganu, keropok lekor is a traditional fish snack or appetizer that showcases the rich culinary heritage of Terengganu and very popular among the locals and tourists. With the combination of fish and sago flour as the main ingredients, the unique texture which has both chewy and slightly crispy especially on the outside makes it gain popularity not only in Terengganu but also throughout Malaysia and neighboring countries like Singapore, Indonesia, Brunei and Thailand. In Malaysia, keropok lekor can be found in various parts of the country such as local markets, food stalls, hypermalls and even through online marketplaces and social media. With the recent impressive market development through online platforms, many keropok lekor producers realized that they need to go beyond traditional method of doing their keropok lekor businesses. However, it is not that simple especially when it involves very large volumes of demands from customers. As reported by study [1], in the production of keropok lekor, most producers are still using traditional manufacturing practices with low competitiveness and poor efficiency. This probably due to the strong belief that adhering to age-old methods will maintain the cultural significance of the keropok lekor. The traditional keropok lekor making process using human touch instead of machine will preserve authentic taste and textures of the keropok lekor [2]. On that account, even to this day, traditional keropok lekor businesses continue to involve a highly labor-intensive and hands-on approach to production and distribution [3].

While certain aspects of the business may have undergone modernization, many producers still steadfastly adhere to traditional methods and recipes. However, this scenario can be further improved through the integration of modern production equipments. Without compromising the traditional taste or texture of keropok lekor, the producers will be able to produce large volumes of demands from their customers. Aside from that, to achieve stability and enhance scalability, it is crucial for keropok lekor producers to have a specific automated solution that will help them streamline their business operations, enhance existing skills, manage business enterprise more systematically while at the same time simplify the business processes, and market their products more aggressively. In other words, there is a need to employ a standard processing procedure to maintain the quality while meeting consumer demands for safety, quality and nutritional value of keropok lekor.

Optimizing a traditional business like Keropok Lekor requires finding a balance between age-old methods and modern technology. For this purpose, an Enterprise Resource Planning (ERP) based system seems to fit the bill. ERP variable is the coordinated information system, which can moreover be called a software bundle with an essential function to integrate all core functions in a company, in any case of the field of the industry [4]. ERP-based system was first introduced in 1960s to cater for only inventory management and control in manufacturing industry. However, it has evolved and become more strategic as the technology has gotten better. In 1990s, modern ERP applications emerged that integrate enterprise-wide backend processes by making use of a computerized sys-

tem. According to study [5], ERP systems are most frequently employed in the business setting for document processing, but they can also be utilized to manage information flow within an organization more effectively. Organization utilizes these systems because they have modules tailored to each function inside a business, enabling the right centralization of data.

II. Related Works

A qualitative study conducted by [6], involving an interview with a renowned keropok lekor producer in Terengganu, revealed that raw material selection, particularly fish type, significantly influences customers' buying preferences. Sardines (tamban and selayang) were identified as the most preferred fish for taste. The study also emphasized the crucial role of effective vacuum packaging in preserving keropok lekor freshness. While this interviews provide valuable qualitative insights into the experiences and opinions of keropok lekor business owner, it is limited by potential biases. A comprehensive, data-driven solution that enables informed decision-making and strategic planning is essential for efficiently managing keropok lekor business operations, improving processes, optimizing productivity, and enhancing customer satisfaction.

To Ensure the safety of consumers, the quality of keropok lekor is paramount. A study by [7] reported the presence of various microorganisms during different processing stages. However, as this study also suggests, microbiological quality can be improved by using raw materials with lower initial microbial loads, employing effective heat treatment during boiling, and implementing proper cooling procedures for the final product. In contrast, a more recent investigation by study [8] identified the presence of the multidrug-resistant (MDR) E. coli B10 strain in ready-to-eat (RTE) keropok lekor, demonstrating its ability to survive under both air-packed (AP) and vacuum-packed (VP) storage at a chilled temperature of 40C. This discovery underscores the importance of food safety for RTE keropok lekor, which ROMAns can address through its quality and inventory management modules.

Technological innovations like keropok lekor drying machines aim to benefit keropok lekor producers by enhancing business performance and reducing maintenance costs. Ultimately, this can minimize the production cost of keropok keping [9]. RFI leverages both traditional sun drying techniques and a custom-designed drying machine to optimize keropok keping production. In contrast to [9], their machine utilizes gas cylinder as its primary energy source, thereby minimizing operational expenses. Our proposed ERP system will facilitate the accurate assessment of RFI's machine's performance, as elucidated in the production management section.

To capitalize on growing market demand and secure higher returns, keropok lekor producers must aim to fully utilize their production capacity. A study done by [10] on Maja Baroh enterprise, one of the SMEs in keropok lekor Industries has stressed on the method of increasing the production size. They need to improve their production floor method whereby some machines or equipments especially the boiling stove need to be replaced to adapt into lean manufacturing system. As for RFI, they seek to expand their production capacity to a target range of 6 to 10 tons per day.

The employee turnover rate in keropok lekor industry, particularly within the production department is notably high. This elevated turnover is significantly influenced by the unpredictable nature of the keropok lekor market demand. The fluctuating and often random demands of this market directly impact the workload and operational tempo within keropok lekor's production facilities. During periods of high demand, production staff may face increased pressure, longer working hours, and potentially more strenuous tasks to meet the surge in orders for their products [6]. Conversely, during lulls in market demand, there might be periods of reduced activity, which could lead to feelings of instability or under utilization among the workforce.

III. ROMAns Framework

An ERP system offers the optimal solution for automating RFI's keropok lekor business operations by providing centralized data management, streamlined processes, integrated functions, real-time insights and reporting, and ensuring regulatory compliance. Fig. 1 illustrates the stages of the system development process for ROMAns. Essentially, the ROMAns ERP system development project for RFI employs a hybrid approach combining waterfall and prototyping methodologies. The overall project adheres to a waterfall model, except for the system development phase, which utilizes prototyping. Integrating prototyping within the Waterfall methodology leverages the strengths of both approaches. Prototyping helps refine requirements and ensure user satisfaction, while Waterfall provides a structured and systematic framework for development. This hybrid approach can lead to a more effective and user-centered ERP system for RFI keropok lekor business operations.

To gain a comprehensive understanding of RFI's keropok lekor business operations, a meticulous preliminary investigation was undertaken. This involved conducting in-depth interviews with key personnel across departments, a thorough examination of existing documentations detailing their workflows and procedures, and direct observation of their current manual processes on the ground in their factory located at Kuala Nerus, Terengganu. The goal of this initial phase was to capture the nuances of their daily activities, identify existing bottlenecks, and understand the specific information flow within the company. This foundational understanding was deemed critical to ensure that any subsequent technological intervention would be truly aligned with RFI's unique operational context and their expected enhancement requirements.

Following this initial fact-finding mission, a comprehensive feasibility study was conducted. This study delved into the technical, operational, and financial implications of implementing an Enterprise Resource Planning (ERP) system within the keropok lekor business. The technical aspect explored the suitability of available technologies and the potential for integration with existing infrastructure, if any. The operational analysis focused on how the proposed ERP features could streamline and enhance RFI's core business processes, from raw material procurement to the final sale of their keropok lekor. Crucially, the financial component assessed the costs associated with the ERP implementation against the anticipated benefits, ensuring a viable return on investment for RFI. This feasibility study served as a critical checkpoint, ensuring that

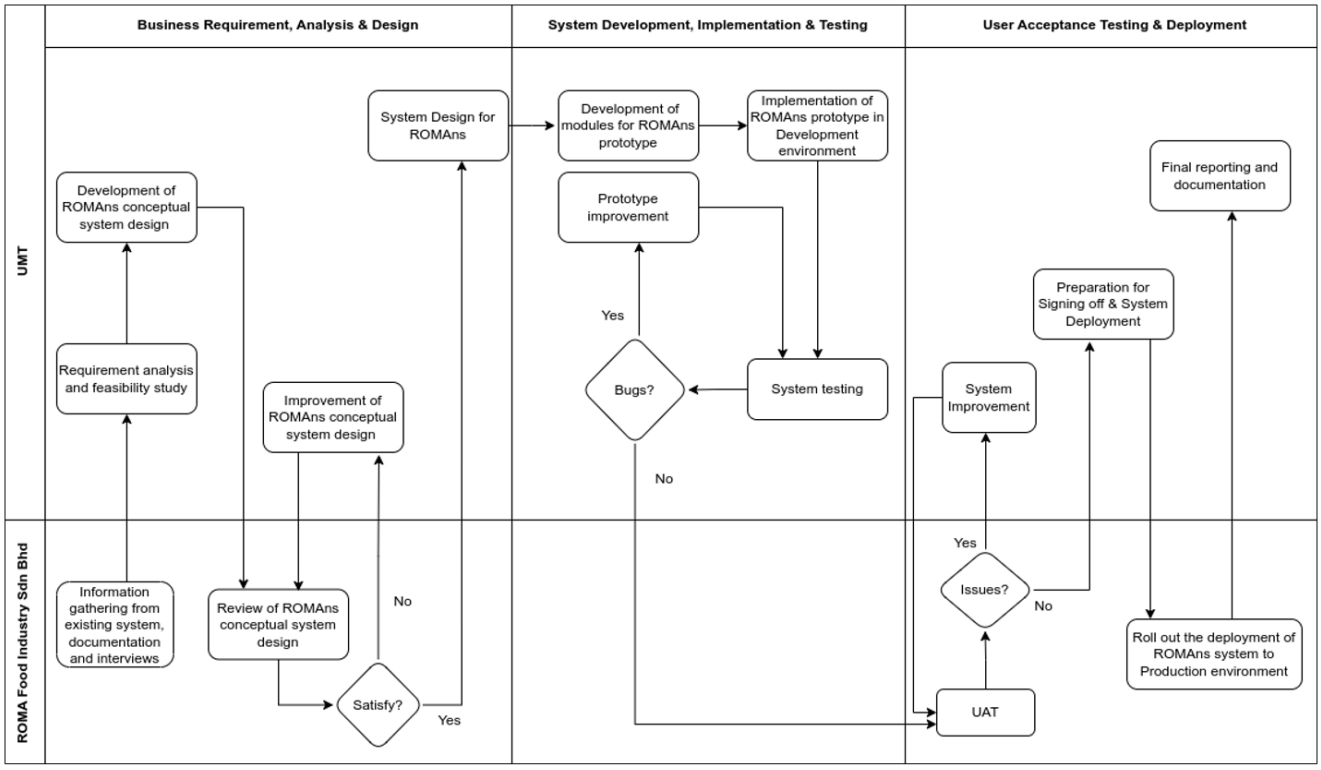


Fig. 1. ROMAns system development.

the proposed ERP solution, which would eventually be named ROMAns, was not only technically sound but also operationally practical and financially justifiable, directly addressing the specific needs and aspirations of the company.

Subsequently, the conceptual design of ROMAns was meticulously modeled. This involved translating the insights gained from the preliminary investigation and the feasibility study into a tangible blueprint of the proposed system. This conceptual model outlined the key functionalities, data flows, and user interfaces of ROMAns. Recognizing the importance of ensuring the system truly met RFI's requirements, this conceptual design underwent an iterative review process. RFI's key personnel were actively involved in examining the proposed model, providing feedbacks, and suggesting modifications until they were completely satisfied that ROMAns accurately reflected their business needs and operational realities. This collaborative approach ensured that the final system design was not just a technical solution but a tool tailored to empower RFI's growth and efficiency within the competitive keropok lekor market of Terengganu.

Once the conceptual design of ROMAns received the full endorsement of RFI, the final system design was constructed. This detailed blueprint served as the foundational architecture for the subsequent development of all identified system modules. This stage involved specifying the technical specifications, data structures, and integration points between different components of ROMAns. The design laid the groundwork for a cohesive and integrated system capable of managing various aspects of RFI's operations. As a result of this thorough planning and design phase, five core modules were identified as being absolutely crucial for the effective management of RFI's keropok lekor business operations. These fundamental modules are: Human Resource (HR), Production, Finance, Sales, and Inventory. Each of these core modules is further comprised of several vital submodules. These submodules are specifically designed to provide granular support for RFI's diverse keropok lekor business processes and operational workflows, as visually represented in Fig. 2, offering a clear overview of the system's modular structure and interdependencies.

1) Human resource management: The volatility stemming from the external market forces acting upon the keropok lekor industry, contributes to the challenges RFI faces in retaining its production employees. The inconsistency in workload and the perceived lack of stability can make employment at the production site less appealing in the long term for some individuals. Consequently, employees might seek more stable employment opportunities elsewhere, leading to the observed high turnover rates. Addressing this issue is crucial for RFI to maintain consistent production quality, reduce recruitment and training costs associated with frequent staff changes, and foster a more experienced and skilled production team dedicated to crafting their Terengganu-style keropok lekor. Understanding the direct link between the market's unpredictability and employee retention is the first step towards developing strategies to mitigate this challenge and create a more stable and attractive working environment within RFI's production operations. Therefore, ROMAns provides Human Resource (HR) module comprises of five significant submodules for smooth resource allocation: Employee Profile (to record and maintain employee details), Schedule (to systematically allocate manpower based on workloads and market/customer demand), Payroll (to record and

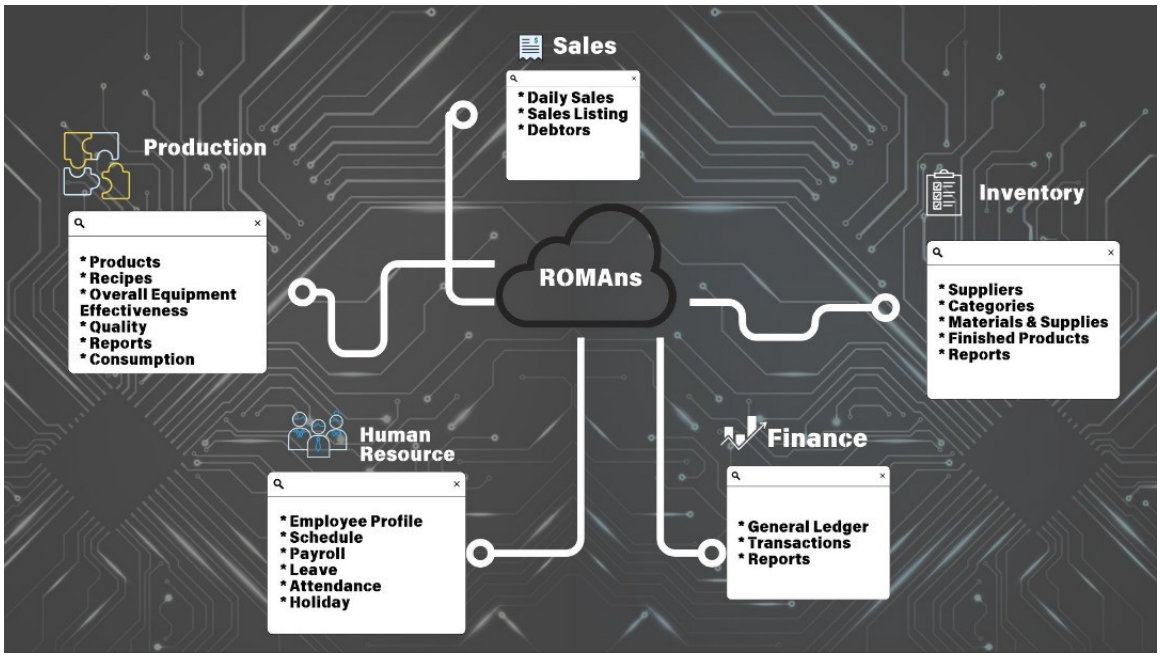


Fig. 2. ROMAns modules.

manage employee salaries), Leave and Attendance (to track and organize employee leaves and attendance), and Holiday (to track national and state public holidays, which are streamlined with staff salaries according to Malaysian employment law).

2) Production management: Recipes are the fundamental blueprint of any food business, crucial for maintaining consistent quality, inspiring culinary creativity and captivating customer loyalty. ROMAns incorporates a recipe submodule to facilitate comprehensive documentation of RFI?s keropok lekor recipes. This module ensures product consistency, enables precise cost control through accurate ingredient quantification, minimizes waste, and optimizes inventory management, ultimately driving their business profitability.

The production module incorporates a quality management submodule to establish and document keropok lekor product standards. Regular customer feedback analysis drives product improvements to align with evolving preferences. In case of quality issues, detailed records are maintained, investigated by the floor production manager, and addressed with corrective actions. The RFI CEO reviews and approves resolutions, ensuring consistent product quality, which is essential for building and maintaining customer trust and loyalty.

RFI utilizes six pieces of equipment or machine in the production of keropok lekor: 1) fish deboner, 2) ingredient mixer, 3) meat mincer, 4) burner, 5) vacuum and 6) drying machine. The productivity of these manufacturing machines can be effectively assessed using Overall Equipment Effectiveness (OEE) metrics, a widely adopted industry standard introduced by Seiichi Nakajima in the 1960s [11]. OEE provides a clear benchmark for evaluating equipment performance, identifying inefficiencies such as machine downtime, defects, and underperformance. Therefore, in this study, the OEE submodule was designed to evaluate overall performance of the utilization of these machines, assess daily keropok lekor production, and increase output while reducing costs related to downtime, repairs, and defective products. Using ROMAns, OEE percentage can be calculated automatically based on the following formula:-

$$\text{OEE\%} = \frac{\text{Actual Output}}{\text{Target Output}} \times 100\% \tag{1}$$

To determine the Target Output(TO), ROMAns will first calculates the Actual Hour (AH) using this equation:

$$\text{AH} = \text{PH - IH - RH - DT} \tag{2}$$

Where

IH = Idle Hour, RH = Rest Hour, DT = Down Time

The Target Output (TO) is then calculated by multiplying the Actual Hour (AH) by the Total Manpower (TM) and the Rate Besen Per Hour (RBHR):

$$\text{TO} = \text{AH x TM x RBHR} \tag{3}$$

A besen holds sixteen kilograms of keropok lekor dough.

Therefore, the final equation to measure the OEE percentage using ROMAns is:

$$\text{OEE} = \frac{\text{AO}}{\text{AH x TM x RBHR}} \times 100\% \tag{4}$$

Meticulous monitoring and management of raw material consumption allow RFI to optimize resource utilization, minimize waste, and enhance product margins. Consumption submodule supports understanding consumption patterns, enabling accurate production cost calculations, informed pricing strategies, and precise profit-loss tracking. Given the reliance on perishable ingredients, effective waste management is crucial for RFI's keropok lekor business. Designed to significantly enhance operational efficiency by identifying and addressing reported waste, analyzing these wastes can provide valuable insights into production inefficiencies, leading to more accurate inventory levels and ultimately improving overall production efficiency and cost analysis. At RFI, waste can occur with both raw materials and finished products. ROMAns automatically tracks this, flagging a high waste event if the combined daily waste exceeds 3% of production. This triggers an immediate alert to RFI Management, prompting them to address the issue in current production or implement preventative measures for future batches. Additionally, the quality of both raw materials and finished products is recorded to assess their impact on production, sales, and the company's overall financial health.

3) Financial management: The general ledger serves as the fundamental backbone of any organization's financial management framework, meticulously documenting every financial transaction that occurs within the business. These transactions are systematically categorized and recorded under specific accounts, providing a comprehensive and auditable trail of the company's financial activities. Currently, RFI's approach to managing their critical financial data involves the manual entry and upkeep of all business transactions within Microsoft spreadsheets. While this method may have served its purpose in the past, it presents limitations in terms of efficiency, accuracy, and the ability to seamlessly integrate financial data with other operational aspects of the business. Recognizing the inherent constraints and potential for errors associated with manual spreadsheet management, the ROMAns system introduces a significant upgrade through its dedicated General Ledger, Transactions, and Reports submodules. These integrated submodules are specifically engineered to provide a more systematic and robust solution for organizing and managing the entirety of RFI's financial transactions. By automating the recording and categorization of financial data, ROMAns aims to enhance accuracy, reduce manual effort, and provide real-time insights into the RFI's financial health. Furthermore, these financial submodules are designed with seamless interlinking capabilities, ensuring that financial data is readily accessible and integrated with all other pertinent modules within the ROMAns system, offering a holistic view of RFI's business performance.

4) Sales management: ROMAns's sales module is designed to comprehensively manage RFI's sales activities, accommodating their practice of allowing both cash and credit payments. To achieve this, the module incorporates three key submodules: Daily Sales Transaction, Sales Listing, and Debtor Details. The Debtor Details submodule is particularly crucial given that a significant portion of RFI's clientele comprises agents and resellers who typically purchase keropok lekor in substantial quantities, often on credit terms. The Daily Sales Transaction submodule facilitates the recording of each individual sales event, while the Sales Listing provides an overview of all sales activities within a specified period. Together, these submodules are pertinent for RFI to effectively track all types of business and sales transactions, monitor outstanding payments from their reseller network, and gain a clear understanding of their sales performance. This structured approach ensures accurate record-keeping and supports informed decision-making regarding sales strategies and credit management.

5) Inventory management: In Terengganu?s highly competitive market, a systematic inventory management system is crucial for RFI's keropok lekor business not just for stock tracking but also for enhancing operational efficiency, sustainability, and cost control. By minimizing waste (especially for perishable ingredients) and ensuring product freshness through First In First Out (FIFO) method, RFI can maintain quality while meeting customer demand promptly, particularly for its agent and reseller network. To address these needs, ROMAns inventory module was developed to cater for a comprehensive inventory management solution. It tracks and manages data on suppliers, raw materials, supplies, and finished products. This system enables RFI to monitor stock levels accurately, provide real-time information to customers, and make informed business decisions based on reliable data.

IV. System Development and Implementation

Prototyping methodology is employed as the system development approach due to RFI's expectation of immediate system utilization, as well as the incomplete understanding of certain requirements and the potential evolution of some parts of production processes. Using this approach, prototypes of the five core modules, as shown in the Fig. 2, are created individually. In the development environment, crude prototype models for each core module are developed and tested individually. Subsequently, these modules are integrated, and the overall ROMAns system undergoes testing. At this stage, RFI representatives also witness all these tests to understand the system's functionality and provide feedback and suggestions. Once system testing is completed and all bugs are rectified, User Acceptance Testing (UAT) is performed by RFI staff. Iterative improvements are made to the ROMAns system until it meets their needs and requirements. Upon successful UAT, the project moves to the deployment phase.

Most keropok lekor producers in Terengganu are small and medium-sized enterprises (SMEs) that rely on traditional methods for production and business practices. Implementing an ERP system for these companies requires a complete shift from their traditional approach. A big bang implementation strategy is suitable due to its shorter implementation timeframe [12]. As bold and ambitious strategy, big bang approach allows simulataneous switch from existing system to the new ERP across departments, functions and processes in a rapid, time saving and cost effective manner. Deploying the ERP through the big bang approach offers easy coordination of streamlined processes that may help RFI achieve quicker returns on investment, easier integration and reporting, reduced running costs as legacy systems are retired simultaneously, a faster implementation timeline, and allows the entire company to benefit from the deployed system by enforcing process changes.

V. Conclusion

ERP plays a crucial role in driving business digitalization and will significantly impact keropok lekor business operations by transforming traditional processes into more efficient, data-driven workflows. Built with Laravel Framework, PHP, and JavaScript, ROMAns is particularly significant for SMEs like RFI, enabling them to consolidate data into a single database for real-time access to accurate information vital for their keropok lekor business processes and effective decision-making. By embracing automation through ROMAns, RFI can enhance its competitiveness in the keropok lekor industry, optimize operations, adapt to market changes, deliver high-quality products, and improve customer service, ultimately leading to sustainable business growth. These advantages contribute to the overall success and competitiveness of the business in today's dynamic landscape.

Acknowledgment

This work is funded by Universiti Malaysia Terengganu (UMT) and ROMA Food Industry Sdn Bhd (RFI) under the Private Partnership Research Grant (PPRG), Vot no. 55443.

References

[1] M. Omar, N. M. Adzahan, H. M. Ghazali, R. Karim, and M. S. B. Karim, "Sustaining traditional food: Consumers' perceptions on physical characteristics of keropok lekor or fish snack," *International Food Research Journal*, vol. 18, pp. 117–124, 2011.

[2] W. N. N. W. M. Hatta, "The authentic of keropok lekor process," *Art and Design Studies*, vol. 27, pp. 1–6, 2015.

[3] M. R. A. Bakar and M. G. Hanafiah, "Business of keropok lekor the tradition of malay terengganu," *Jurnal Wacana Sarjana*, vol. 2, no. 3, pp. 1–7, 2018.

[4] U. Rahardja, "Implementation of Enterprise Resource Planning (ERP) in Indonesia to increase the significant impact of management control systems," *APTISI Transactions on Management*, vol. 7, no. 22, pp. 152–159, 2022.

[5] L. E. L. Barna and V. A. Enachescu, "ERP system application in the education environmental," in *European Proceedings of Educational Sciences*, Emanual, Ed., vol. 5, no. 1. European Publisher, April 2023, pp. 537–542. [Online]. Available: https://doi.org/10.15405/epes.23045.55

[6] A. S. Suhaimi, W. I. M. W. Lokman, W. A. M. W. Abdullah, and N. A. N. Ariffin, "Makanan Warisan Terengganu (Keropok Warisan Losong)," *Jurnal Sejarah Lisan Malaysia*, vol. 5, no. Isu Khas, pp. 280–300, 2021.

[7] N. K. M. A. Rashid, Z. Hassan, J. Bakar, and G. R. R. Ali, "Microbiological quality of keropok lekor during processing," *International Food Research Journal*, vol. 16, pp. 215–223, 2009.

[8] W.-H. H, L. MN, H. Y, A. R, H. Z, and M. NA, "Survival of antibiotic resistant *Escherichia coli* in vacuum packed keropok lekor: Food safety alert among SME keropok lekor producers." *Tropical Biomedicine*, vol. 37, no. 1, pp. 103–115, 2020.

[9] M. S. Z. M. Suffian, S. Mohamaddan, A. M. A. A. M. Mohtar, and M. N. Leman, "Modal analysis of keropok keping drying machine," in *IOP Conference Series: Materials Science and engineering*, vol. 1062, no. 1. IOP Publishing, Feb 2021, p. 012004. [Online]. Available: https://dx.doi.org/10.1088/1757-899X/1062/1/012004

[10] F. Ayob, M. K. A. M. Ariffin, and S. Sulaiman, "Development of Boiling Stove for Lekor Ball's Production; Local Industries (SME)," in *Proceeding International Multidisciplinary Conference (IMC 2022)*. UNIMAP, 2022, pp. 239–250.

[11] B. Setiawan, F. A. Latif, and E. Rimawan, "Overall equipment effectiveness (OEE) analysis: A case study in the PVC compound industry." *Indonesian Journal of Industrial Engineering & Management*, vol. 3, no. 1, pp. 14–25, 2022.

[12] E. A. Duplaga and M. Astani, "Implementing ERP in manufacturing," *Information System Management*, vol. 20, pp. 68–75, 2003.

Enhanced Phishing Website Detection Using Optimized Ensemble Stacking Models

Zainab Alamri[1], Abeer Alhuzali[2], Bassma Alsulami[3], Daniyal Alghazzawi[4]
Department of Computer Science-Faculty of Computing and Information Technology,
King Abdulaziz University, Jeddah, Saudi Arabia[1,2,3]
Information Systems Department-Faculty of Computing and Information Technology,
King Abdulaziz University, Jeddah, Saudi Arabia[4]

***Abstract*—Phishing attacks remain a persistent and evolving cybersecurity threat, necessitating the development of highly accurate and efficient detection mechanisms. This research introduces an optimized ensemble stacking framework for phishing website detection, leveraging advanced machine learning techniques, hybrid feature preprocessing, and meta-learning strategies. The proposed approach systematically evaluates nine diverse base classifiers: XGBoost, CatBoost, LightGBM, Random Forest, Gradient Boosting, Extra Trees, Support Vector Classifier, AdaBoost, and Bagging. We compare baseline classifiers, a standard ensemble stacking model, and four optimized stacking configurations across four balanced and imbalanced datasets. Our optimized ensemble stacking achieves perfect accuracy (one hundred percent) on the first two datasets, and over ninety-nine percent accuracy on the two more challenging imbalanced datasets. A direct comparison with related studies demonstrates that our optimized stacking approach delivers superior detection accuracy.**

***Keywords*—Phishing detection; machine learning; ensemble stacking; cybersecurity**

I. Introduction

The increasing reliance on online services has made individuals and organizations more vulnerable to cyberattacks, with phishing remaining a significant threat [1]. Phishing attacks employ deceptive techniques to steal sensitive information, often masquerading as legitimate entities to trick users into divulging credentials or financial details [2]. Despite existing security measures, phishing continues to evolve, posing persistent challenges to both individuals and organizations [1]. As of 2024, the rate of phishing attacks remains alarmingly high, with reports indicating that the number of phishing attacks detected worldwide ranges from hundreds of thousands to millions each month [3].

Traditional phishing detection systems often rely on machine learning algorithms and manually crafted features [1]. However, these systems struggle to keep pace with the constantly evolving tactics employed by phishers [4]. While various machine learning, deep learning, and other approaches have been proposed, their detection accuracy needs further improvement [5]. Ensemble learning, particularly stacking (stacked generalization), has emerged as a promising technique for enhancing the performance of classification models by combining the strengths of multiple base classifiers [6]. Stacking involves training a meta-classifier on the outputs of individual base classifiers, potentially leading to a more robust and accurate model [7]. In 2024, the proliferation of sophisticated phishing websites poses a significant threat to individuals and organizations, resulting in substantial financial losses and personal data breaches [1]. Despite numerous detection methods, including machine learning and deep learning techniques, the accuracy and generalizability of these approaches remain insufficient to effectively combat the evolving tactics of cybercriminals [4]. The dynamic nature of phishing attacks, characterized by the use of advanced evasion techniques and rapidly changing features, necessitates the development of more robust and adaptive detection systems. Traditional phishing detection systems that rely on machine learning and manual features struggle with evolving tactics [8].

Phishing website detection has become a critical area of research due to the increasing sophistication and frequency of phishing attacks targeting individuals, organizations, and even IoT environments. Machine learning (ML) and deep learning methods, particularly ensemble learning techniques such as stacking, bagging, and boosting, have shown promise in improving detection accuracy by leveraging the strengths of multiple classifiers. Recent studies have demonstrated that stacking ensemble models, especially when optimized, can outperform single classifiers and traditional detection methods in terms of accuracy, recall, and other performance metrics[9], [10]. Despite these advances, several limitations persist in current research. Many studies rely on a fixed set of base classifiers without systematically selecting or optimizing the most effective algorithms for stacking, which can potentially limit model performance [11], [12], [13]. Optimization efforts, when present, often focus on parameter tuning for individual models rather than holistic selection and combination of diverse, high-performing classifiers [5], [11], [14]. Additionally, some approaches lack robust validation across multiple, diverse datasets, raising concerns about the generalizability of their results to real-world scenarios [15], [9], and [5]. There is also a need for more comprehensive feature selection and integration strategies to further enhance detection capabilities [15], [14]. These gaps are significant because suboptimal model selection, insufficient optimization, and limited generalizability can result in lower detection accuracy and increased vulnerability to evolving phishing tactics. Inadequate detection systems may fail to protect users and organizations from financial loss, data breaches, and reputational damage, especially as attackers continually adapt their methods to bypass existing defenses [9], [16]. This study addresses the research question: How can optimized ensemble stacking improve phishing website detection accuracy across diverse datasets with varying sizes,

balances, and feature complexities? To answer this, we systematically select and optimize a diverse set of strong base models for stacking, employ advanced optimization techniques, and validate the enhanced ensemble stacking approach across multiple datasets. By focusing on both algorithm selection and parameter optimization, the proposed method achieves higher detection accuracy, improved robustness, and better adaptability to new phishing strategies.

This study responds to these challenges by systematically selecting and optimizing a diverse set of strong base models for stacking, employing advanced optimization techniques, and validating the enhanced ensemble stacking approach across multiple datasets. By focusing on both algorithm selection and parameter optimization, the proposed method aims to achieve higher detection accuracy, improved robustness, and better adaptability to new phishing strategies, thereby strengthening real-world cybersecurity defenses. By extensively comparing various ensemble configurations, this research demonstrates the effectiveness of advanced ensemble stacking techniques in identifying phishing websites with high precision, reliability, and efficiency.

This study provides the following contributions:

- Develop an optimized ensemble stacking model with optimized hyperparameters, preprocessing, and feature engineering for performance improvement.
- Provide ensemble stacking variations for optimal results.
- Evaluate the model comprehensively across multiple datasets for performance assessment.

To achieve the above contributions, we develop an enhanced phishing website detection approach that leverages optimized ensemble stacking models to improve classification accuracy. This method integrates multiple machine learning classifiers within a structured ensemble framework, including Random Forest, Gradient Boosting, XGBoost, CatBoost, LightGBM, and Support Vector Classifier SVC, among others, capitalizing on their complementary strengths. A meta-classifier (Logistic Regression and CatBoost) is used to aggregate the outputs of the base models. Additionally, variations of stacking models are compared across four datasets to determine the optimal configurations. The model is tested on four datasets, each subjected to several variations of the ensemble stacking model to identify the most effective configuration.

By extensively comparing the four *optimized* stacking configurations, this research demonstrates the effectiveness of our advanced stacking techniques in identifying phishing websites with high precision, reliability, and efficiency. As a preview of our principal findings, Optimized Stacking-1 and -2 each achieve **100 %** accuracy on the two clean, balanced benchmarks, while all four variants maintain above **99 %** accuracy on the more challenging imbalanced datasets—underscoring both the robustness and practical deployability of our approach.

The rest of the paper is organized as follows. Section II highlights key related research studies. Sections III, IV, and V explain our methodology, evaluation metrics, and experimental results. Finally, we conclude in Section VII.

II. Related Work

This section reviews relevant research in two areas. First, we examine ensemble learning approaches, particularly stacking, which have shown effectiveness in phishing detection by combining multiple classifiers. Second, we explore optimization techniques that enhance stacking performance, including model selection, hyperparameter tuning, and meta-learner strategies.

A. Ensemble Learning for Phishing Detection

Ensemble methods, which combine multiple individual classifiers, have emerged as a powerful approach to enhance the accuracy and robustness of phishing detection systems. To overcome the limitations of individual classifiers, ensemble methods have been proposed, which combine the predictions of multiple models to improve overall accuracy and robustness [17].

Ensemble methods, while powerful, have certain limitations when applied to phishing detection. One significant limitation is the inability of some ensemble techniques, like random forests, to capture high correlations between features and their joint dependency on the label, which can affect the model's performance in complex datasets [18]. Additionally, traditional ensemble methods may struggle with the evolving nature of phishing attacks, as they often rely on static features that do not adapt well to new phishing tactics [19]. Furthermore, the computational cost and complexity of ensemble methods can be high, which may not be suitable for real-time detection scenarios [19].

Ensemble models typically require more resources for training and prediction, as they involve combining the outputs of multiple base classifiers. This increased computational overhead can be a drawback, especially in scenarios where real-time or low-latency detection is required. Additionally, the complexity of tuning and optimizing ensemble methods may pose challenges, as selecting the appropriate base classifiers and configuring the meta-classifier can significantly impact overall performance [20].

When comparing traditional ensemble methods and stacking ensembles, the latter often demonstrates superior performance in phishing detection tasks. For example, stacking models have achieved higher accuracy and F1 scores compared to standalone ensemble methods, such as random forests or AdaBoost [21]. The ability of ensemble stacking to combine the strengths of multiple algorithms and mitigate their weaknesses makes it a good choice for phishing detection [5], [9].

Stacking addresses these challenges by intelligently combining the strengths of diverse base classifiers, mitigating the risk of overfitting, and improving generalization performance, which can lead to more accurate and robust phishing website detection [17]

B. Optimization Techniques in Stacking Models

Phishing website detection has been a significant area of research due to the increasing sophistication of phishing attacks. Several studies have investigated the application of ensemble machine learning models to improve detection accuracy.

One approach involves the use of optimized stacking ensemble models, which combine multiple machine learning algorithms to improve detection performance. For instance, a study utilized a genetic algorithm to optimize parameters of ensemble methods, including Random Forest, AdaBoost, and XGBoost, achieving detection accuracies of 97.16%, 98.58%, and 97.39% across different datasets [5]. Another study reported a similar approach, achieving a detection accuracy of 97.16% by optimizing ensemble classifiers [22].

Other research has focused on stacking models that integrate multiple classifiers, such as Random Forest, Gradient Boosting, and AdaBoost, with logistic regression as an aggregator. This model achieved an accuracy of 98.72% and demonstrated superior performance compared to individual algorithms [10]. Additionally, a multilayer stacked ensemble learning model achieved accuracies ranging from 96.79% to 98.90% across various datasets, highlighting the effectiveness of layered ensemble techniques [9].

The accuracy of phishing website detection models varies across studies, with several achieving high performance. For example, one study reported an accuracy of 98.72% using a stack ensemble model [10], while another achieved 99.31% with a stacked classifier model employing six algorithms [3]. A different approach using a stacking model with URL and HTML features achieved 97.30% accuracy on one dataset and 98.60% on another [23].

Despite the high accuracy rates reported, there are limitations and gaps in current research. Many studies emphasize the need for further enhancement of detection accuracy and adaptability to evolving phishing tactics [22], [24]. Additionally, while ensemble models show promise, they often require complex optimization and feature selection processes, which can be computationally intensive [15], [25]. There is also a need for real-time detection capabilities, as many models are tested in controlled environments and may not perform as well in dynamic, real-world scenarios [23].

Another study optimized stacking ensemble methods using a Genetic Algorithm to tune parameters of various ensemble methods such as Random Forest, AdaBoost, XGBoost, and others. This approach achieved detection accuracies of 97.16% to 98.58% across different datasets, demonstrating significant improvements over traditional methods [22], [5].

Phishing website detection has been a significant area of research due to the increasing threat posed by phishing attacks. Various studies have explored the use of ensemble models to enhance detection accuracy. A stack ensemble model combining RandomForest, GradientBoosting, and AdaBoost with logistic regression as an aggregator achieved an accuracy of 98.72%, demonstrating superior performance over individual algorithms and existing studies [10]. Another study proposed an optimized stacking ensemble method using a genetic algorithm to tune parameters, achieving detection accuracies of 97.16%, 98.58%, and 97.39% across different datasets [5]. Similarly, a multilayer stacked ensemble learning model reported accuracies ranging from 96.79% to 98.90% across various datasets, outperforming baseline models [9].

Despite the high accuracy rates, several limitations and gaps remain in the current research. First, many models are tested on specific datasets, which may not generalize well to other datasets or real-world scenarios [5], [9]. Second, some models rely heavily on specific features, such as URL and HTML characteristics, which may not be present in all phishing websites [23]. Third, the complexity of ensemble models can lead to increased computational costs, making real-time detection challenging [15], [3]. Fourth, phishing tactics continue to evolve, necessitating ongoing updates and adaptations of detection models to maintain effectiveness [24], [21] . Last, the use of multiple models in stacking can increase the risk of overfitting, particularly if the base models are too complex or if the dataset is not sufficiently large or diverse [9], [5].

In summary, while ensemble models have significantly improved phishing website detection accuracy, challenges remain in optimizing these models for real-time applications and adapting to new phishing strategies. Our work addresses these limitations and designs a more robust and efficient detection system.

III. Methodology

This research employs a systematic and structured methodology to enhance the detection of phishing websites by implementing optimized ensemble stacking models. The proposed approach, aimed at improving phishing website detection, is uniformly applied across four benchmark datasets. It follows a systematic sequence involving data preprocessing, model training, and performance evaluation, and is consistently implemented on the following datasets: Dataset 1 [26], Dataset 2 [27],and Dataset 3 and Dataset 4, both derived from the same source [28]. We ensure consistency in base model selection (RandomForest, XGBoost, CatBoost, GradientBoosting, LightGBM, ExtraTrees, AdaBoost, Bagging, and SVC) across all datasets. However, ensemble stacking models are adjusted in hyperparameters, iterations, and cross-validation strategy for each dataset to balance execution time and accuracy. The methodology incorporates data preprocessing techniques, including feature encoding, SMOTE oversampling for imbalanced datasets, and feature standardization. The evaluation framework uses accuracy, classification reports, and confusion matrices to compare performance across datasets and determine the most effective dataset for phishing detection.

The proposed framework involves several steps, which are summarized below:

- Perform necessary preprocessing steps, such as encoding categorical variables, scaling numerical features, and addressing class imbalances using SMOTE on the selected datasets.
- Train and evaluate multiple individual classifiers, including RandomForest, XGBoost, CatBoost, LightGBM, GradientBoosting, ExtraTrees, SVC, AdaBoost, and Bagging, for each dataset. This establishes a baseline performance for comparison before implementing stacking ensembles. The results include accuracy scores and classification reports for all datasets.
- The structure of the stacking ensemble involves training a traditional stacking ensemble model using a variety of base classifiers, including XGBoost, Random Forest, CatBoost, LightGBM, Gradient Boost-

ing, Extra Trees, SVC, AdaBoost, and BaggingClassifier, with Logistic Regression as the final estimator. The improved stacking ensemble model incorporates optimized hyperparameters, balancing techniques (SMOTE), and boosting-based feature selection implicitly through models like XGBoost, CatBoost, and LightGBM. The improved version uses CatBoost as the final estimator and employs StratifiedKFold cross-validation to enhance generalization.

- Evaluate and compare the performance of individual base models, traditional stacking ensemble, and improved stacking ensemble across all datasets. Metrics such as accuracy, precision, recall, F1-score, and confusion matrices will be used for comparison.
- Illustrate the performance across models and datasets using various visualization methods. These visual aids help identify trends, strengths, and weaknesses in the models for each dataset.

The primary objective of this study is to investigate how variations in dataset characteristics impact the performance of the improved ensemble stacking model and to identify which dataset yields the most effective results. The overall workflow of the proposed approach is illustrated in Fig. 1.

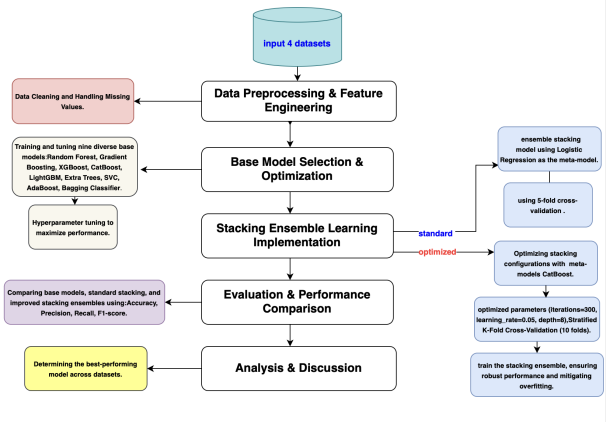


Fig. 1. Overview of the proposed ensemble stacking workflow.

A. Dataset Description

This study utilizes four datasets of varying sizes, features, and class distributions to train, test, and evaluate the phishing website detection model. These datasets ensure a comprehensive assessment of model performance under different conditions. Balanced datasets facilitate the assessment of baseline performance without introducing bias, thereby providing a clear foundation for evaluating model accuracy. In contrast, unbalanced datasets are used to test the robustness of the models, with SMOTE applied to generate synthetic samples and improve classification fairness. This combination of balanced and unbalanced datasets enabled a comprehensive evaluation of the models' robustness and their practical applicability in real-world phishing detection scenarios.

1) Balanced datasets: include Dataset 1 [26] (11,430 instances, 89 features), titled Web Page Phishing Detection and sourced from Mendeley Data, and Dataset 2 [27] (11,481 instances, 89 features), also titled Web Page Phishing Detection and obtained from Kaggle. Both datasets contain an equal number of phishing and legitimate website samples. Specifically, Dataset 1 includes 5,715 phishing and 5,715 legitimate samples, while Dataset 2 consists of 5,740 phishing and 5,741 legitimate samples. These balanced distributions eliminate class bias and support reliable model evaluation without the need for resampling techniques. Both datasets feature a rich set of attributes relevant to phishing detection, such as length_url, nb_dots, domain_age, iframe, tld_in_path, google_index, and ratio_digits_host. These features capture URL structure, domain trustworthiness, and content behavior, which are critical for distinguishing phishing from legitimate websites.

2) Unbalanced datasets: include Dataset 3 [28] (88,647 instances, 112 features) and Dataset 4 [28] (58,645 instances, 112 features), both derived from the Phishing Websites Dataset available on Mendeley Data, both of which exhibit varying degrees of class imbalance. Dataset 3 contains 30,648 phishing and 57,999 legitimate samples (34.57% phishing,65.43% legitimate), while Dataset 4 includes 30,651 phishing and 27,994 legitimate samples (52.26% phishing, 47.74% legitimate). To address this imbalance, the Synthetic Minority Oversampling Technique (SMOTE) was applied. These datasets include more advanced and diverse features such as ttl_hostname,qty_redirects, url_shortened, tls_ssl_certificate, and domain_google_index, which reflect redirection patterns, SSL usage, and DNS behaviors.

B. Data Preprocessing

To ensure consistency and accuracy, a structured preprocessing pipeline was applied across all datasets. First, non-numeric columns were identified, and missing values in categorical features were replaced with the string "Missing" to retain the information. Second, LabelEncoder was used to convert categorical features and the target variable into a numerical format for compatibility with machine learning models. Third, SMOTE was applied with an 80% sampling strategy to balance minority and majority classes, improving model generalization. Each dataset was split into 80% training and 20% testing, ensuring fair evaluation on unseen data. Lastly, StandardScaler transformed the features to have a mean of 0 and a standard deviation of 1, benefiting models that are sensitive to feature magnitude. All four datasets underwent identical preprocessing steps to maintain fairness in performance comparisons. By applying these steps, the data was cleaned, balanced, and optimized, ensuring reliable training and evaluation of the phishing detection models.

C. Selected Base Models and Ensemble Learning Design

To ensure robust and generalizable phishing detection, a diverse set of nine base classifiers was selected and consistently applied across four benchmark datasets: Dataset 1, Dataset 2, Dataset 3, and Dataset 4. These models were chosen to leverage complementary strengths in handling non-linear patterns, categorical features, boosting mechanisms, and high-dimensional data.

The base models used in all experiments include gradient boosting algorithms (XGBoost, CatBoost, LightGBM,

GradientBoostingClassifier), ensemble tree methods (RandomForestClassifier, ExtraTreesClassifier), kernel-based classifiers (SVC), and ensemble meta-learners such as AdaBoostClassifier and BaggingClassifier. Each model is known for its strong performance in phishing detection and tabular classification problems [29], [30], [31], [8]. The gradient boosting algorithms are known for their high accuracy and ability to capture complex relationships [8]. XGBoost is also efficient and scalable. CatBoost is particularly good at handling categorical features. The ensemble tree-based models can handle complex, nonlinear feature interactions well and have demonstrated high accuracy in phishing detection tasks [29], [30], [32] . Support Vector Classifier (SVC) with nonlinear kernels captures subtle lexical and structural patterns in URLs, contributing to high classification accuracy and generalization across datasets [30], [31]. AdaBoost enhances the performance of weak learners by focusing on complex samples, thereby improving overall detection accuracy [30]. Bagging reduces variance by aggregating multiple models trained on different data samples, improving robustness and generalization [31].

Comparative studies highlight the effectiveness of combining diverse classifiers, including SVC, AdaBoost, and Bagging, in the phishing detection task [30]. These diverse machine learning models, including ensemble tree-based methods, gradient boosting algorithms, kernel-based classifiers, and ensemble stacking techniques that reduce variance or improve weak learners, have been effectively used together to detect phishing websites with high accuracy [33], [19], [4].

1) Training and evaluation of models: To systematically assess model performance, a dedicated function was designed to train and evaluate multiple classifiers on the phishing datasets. This function accepts training and testing subsets for each dataset and outputs standardized evaluation metrics for comparative analysis. The procedure involves training each classifier on the training data, generating predictions on the test data, and evaluating performance using key classification metrics. Each model is evaluated on all four datasets to examine its generalizability across varying data characteristics. By applying this process uniformly across balanced and unbalanced datasets, it was possible to identify base models that consistently deliver high performance.

Specifically:

- Models were trained and evaluated on Dataset 1 and Dataset 2, both of which are balanced.
- The same process was applied to Dataset 3 and Dataset 4, which are imbalanced and larger in scale.

For each dataset, model results were stored and analyzed using:

- Accuracy: Overall classification accuracy for each model.
- Classification Report: Detailed metrics including precision, recall, and F1-score per class.

D. Standard Ensemble Stacking Approach

The ensemble stacking approach integrates the predictive capabilities of multiple diverse base classifiers into a unified model, aiming to significantly enhance phishing website detection accuracy. This layered architecture consists of a base layer where classifiers are trained independently, and a meta-layer that learns to optimally combine their predictions into a final decision. The effectiveness of Logistic Regression as a meta-learner in stacking ensembles has been demonstrated in phishing detection systems, achieving 98% accuracy when combined with base classifiers like Random Forest and XGBoost [34]. To ensure generalization and reduce overfitting, 5-fold cross-validation was employed during model training, as recommended in recent phishing detection research [35].

In this study, a consistent standard stacking framework was applied across our four datasets, including the two balanced datasets. This structural consistency enables a fair comparison and allows analysis of how dataset characteristics affect ensemble performance. The base layer comprised nine machine learning classifiers selected for their proven effectiveness and complementary strengths: RandomForest, Gradient Boosting, XGBoost, CatBoost, LightGBM, ExtraTrees, SVC, AdaBoost, and Bagging. Each classifier was chosen to contribute diverse perspectives in identifying phishing behavior patterns. For this stacking configuration, Logistic Regression was used as the meta-learner to combine the outputs of the base classifiers. To ensure generalization and reduce overfitting, 5-fold cross-validation was employed during model training.

E. Optimized Ensemble Stacking Approach

To improve the accuracy and efficiency of phishing website detection, we conduct four variations to optimize the ensemble stacking models. These models are systematically refined to strike a balance between complexity, computational efficiency, and detection performance.

- Optimized Stacking-1: comprehensive stacking model. This version employs a comprehensive ensemble stacking approach that integrates the nine base classifiers, including base classifiers RandomForest, XGBoost, CatBoost, LightGBM, Gradient Boosting, Extra Trees, SVC, AdaBoost, and Bagging. The CatBoost classifier was utilized as the meta-model, with hyperparameter tuning applied to optimize performance.
- Optimized Stacking-2: performance-oriented selection. This architecture prioritizes high-performing models by selecting XGBoost, CatBoost, and RandomForest as base classifiers, thereby reducing computational complexity while maintaining strong classification capabilities. The CatBoost classifier is used as the meta-model.
- Optimized Stacking-3: efficiency-optimized stacking. To enhance computational efficiency, the number of base models is reduced, retaining only the most effective classifiers: XGBoost, CatBoost, RandomForest, LightGBM, Gradient Boosting, and SVC. The CatBoost classifier continued to serve as the meta-model, ensuring robustness in predictions.
- Optimized Stacking-4: minimalist high-performance model. Designed for optimized execution and minimal computational cost, this model leverages a CatBoost classifier with advanced hyperparameter tuning as

the meta-model. The base models are reduced to XGBoost, CatBoost, RandomForest, LightGBM, and Gradient Boosting, achieving high efficiency while preserving detection accuracy. This ensemble model achieves a balance between execution time and detection accuracy.

1) Design and configuration of the optimized ensemble stacking: Each stacking configuration followed a unified pipeline. Non-numeric features are encoded using LabelEncoder, and class imbalance is addressed using SMOTE. Feature scaling is performed using StandardScaler to ensure consistency across numerical values. Stratified K-Fold cross-validation is used for training: 10-fold for balanced or smaller datasets (Datasets 1, 2, and 4), and 5-fold for the larger dataset (Dataset 3) to reduce execution time. A diverse set of nine base classifiers is employed to enhance generalization and capture a broad range of learning patterns: tree-based models (RandomForestClassifier, ExtraTreesClassifier, and GradientBoostingClassifier), boosting techniques (XGBClassifier, CatBoostClassifier, LGBMClassifier, and AdaBoostClassifier), a kernel-based model (SVC), and a bagging-based method (BaggingClassifier).

2) Base model optimization: The same base model structure is applied across all datasets, but hyperparameters are adjusted to match the dataset size and complexity. For the balanced datasets (Datasets 1 and 2), base models are configured with higher complexity. For the large, imbalanced datasets (Datasets 3 and 4), parameter values are reduced to accelerate training. Specifically, we perform the following key adjustments:

- Number of Estimators: 500 estimators for Datasets 1 and 2; reduced to 100 or 50 for Datasets 3 and 4.
- Learning Rate: It is set to 0.05 for smaller datasets; increased to 0.1 for larger datasets.
- Depth: Models used greater depth (10–25) for balanced data; shallower depth (6–10) for large-scale datasets in Optimized Stacking-1 and Stacking-3.

It is important to note how hyperparameter settings influenced performance across datasets. Increasing the number of estimators (e.g., 500 on Datasets 1 and 2) improved accuracy and reduced variance, but also increased training time, which is why smaller values (100 or 50) were used for the larger datasets. Similarly, greater tree depth (10–25) enhanced recall by capturing complex phishing patterns, but shallower depths (6–10) were more efficient for large-scale datasets and reduced the risk of overfitting. Adjusting the learning rate also played a critical role: lower values (0.05) improved model stability and recall on imbalanced data, while higher values (0.1) accelerated convergence but risked losing fine-grained detection capability. For the CatBoost meta-model, more iterations (300) strengthened performance on smaller balanced datasets, whereas fewer iterations (50–100) were sufficient for larger datasets to maintain efficiency without significant accuracy loss. These observations confirm that hyperparameter tuning was essential not only for maximizing accuracy but also for balancing robustness and computational efficiency across datasets.

3) Meta-model optimization: The meta-model used in all stacking ensembles is `CatBoostClassifier`, replacing Logistic Regression to improve generalization and robustness. The CatBoostClassifier demonstrates strong performance due to its efficient handling of categorical features and robustness on imbalanced datasets, which are commonly encountered in phishing detection scenarios [25]. Configurations are tailored per dataset:

- Balanced datasets 1 and 2: 300 iterations, depth 8, learning rate 0.05, with 10-fold cross-validation.
- Large and imbalanced Dataset 3:
  - Optimized Stacking-1 and Optimized Stacking-3: 50 iterations, depth 6, learning rate 0.1, with 5-fold CV.
  - Optimized Stacking-2 and Optimized Stacking-4: 300 iterations, depth 8, learning rate 0.05, with 10-fold CV.
- Dataset 4:
  - Optimized Stacking-1: 100 iterations, depth 6, learning rate 0.1, with 10-fold CV.
  - Optimized Stacking-2, Optimized Stacking-3, Optimized Stacking-4: 300 iterations, depth 6, learning rate 0.1, with 10-fold CV.

IV. Models Evaluation

A. Experimental Setup

To evaluate the effectiveness of the proposed optimized ensemble stacking models for phishing website detection, all experiments were executed in Google Colab, a cloud-based Jupyter environment. The programming environment was configured with Python 3 and accelerated using the v5e-1 TPU to handle the computational demands of training complex ensemble models on large datasets. We utilized several libraries such as scikit-learn [36], XGBoost [37], LightGBM [38], CatBoost [39], Matplotlib [40], Seaborn [41], Pandas [42], and NumPy [43].

B. Evaluation Metrics

To thoroughly evaluate the performance of the proposed *optimized ensemble stacking model* for phishing website detection, multiple classification metrics were employed. These metrics provide a comprehensive view of each model's effectiveness, especially in distinguishing between legitimate and phishing websites across datasets with varying distributions.

- Accuracy: Represents the overall proportion of correctly classified instances:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

  However, while accuracy is informative in balanced datasets, it may be misleading in imbalanced scenarios.

- Precision: Measures the proportion of true phishing detections among all predicted phishing cases:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

High precision reduces false alarms and is essential for protecting legitimate websites from misclassification.

- Recall (Sensitivity): Evaluates the model's capability to correctly detect actual phishing sites:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3}$$

High recall ensures phishing threats are not overlooked.

- F1-Score: The harmonic mean of precision and recall, providing a single balanced metric:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

It is especially useful in imbalanced data situations.

- AUC-ROC: Measures the area under the Receiver Operating Characteristic curve, reflecting the model's ability to distinguish between classes independently of the classification threshold. A value close to 1.0 indicates excellent performance.
- Confusion Matrix: Summarizes true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN), offering detailed insights into classification errors.
- Execution Time: Records the time required for model training and evaluation. This is crucial for determining the feasibility of deploying the model in real-time phishing detection systems.

V. Experiments and Results

To evaluate the proposed optimized ensemble stacking models, experiments were conducted on four phishing datasets differing in size, class balance, and feature structure. The same stacking architecture and base models were applied across all datasets, including four optimized stacking (optimized-stacking1 to optimized-stacking4) to ensure consistent comparison. Preprocessing involved label encoding, standard scaling, and SMOTE for imbalanced datasets. For large datasets, model parameters were adjusted to reduce training time without sacrificing accuracy. Evaluation used an 80:20 split and stratified cross-validation, measuring accuracy, precision, recall, F1-score, and confusion matrices to assess performance under different data conditions.

A. Experiment 1: Evaluation of Base Models Individually

Before implementing the stacking ensembles, nine base classifiers were independently evaluated to assess their generalizability and standalone performance across four phishing datasets: Dataset 1, Dataset 2, Dataset 3, and Dataset 4. These models include Random Forest, Gradient Boosting, XGBoost, CatBoost, LightGBM, Extra Trees, Support Vector Classifier (SVC), AdaBoost, and Bagging. Each model was selected based on its distinct learning paradigm and complementary strengths.

All classifiers were trained using an identical experimental pipeline that included preprocessing steps such as categorical encoding, SMOTE for handling class imbalance (in unbalanced datasets), and feature scaling. A fixed random seed was used for reproducibility, and models were evaluated on an 80/20 train-test split. Performance was measured using standard classification metrics including accuracy, precision, recall, F1-score, and AUC. The detailed performance metrics for all base classifiers are presented in Tables I through IV, with corresponding ROC curves illustrated in Fig. 2 to 5.

TABLE I. Base Models' Performance on Dataset 1

Model	Precision	Recall	F1-Score	Accuracy (%)	AUC
Random Forest	0.9695	0.9693	0.9694	96.94	0.9945
Gradient Boosting	0.9593	0.9593	0.9593	95.93	0.9919
XGBoost	0.9733	0.9733	0.9733	97.33	0.9956
CatBoost	0.9729	0.9728	0.9729	97.29	0.9959
LightGBM	0.9720	0.9720	0.9720	97.20	0.9958
Extra Trees	0.9696	0.9692	0.9694	96.94	0.9952
SVC	0.9628	0.9628	0.9628	96.28	0.9930
AdaBoost	0.9536	0.9536	0.9536	95.36	0.9884
Bagging	0.9554	0.9553	0.9554	95.54	0.9873

TABLE II. Base Models' Performance on Dataset 2

Model	Precision	Recall	F1-Score	Accuracy (%)	AUC
RandomForest	0.9808	0.9809	0.9808	98.08	0.9966
GradientBoosting	0.9599	0.9600	0.9599	95.99	0.9915
XGBoost	0.9856	0.9856	0.9856	98.56	0.9976
CatBoost	0.9856	0.9856	0.9856	98.56	0.9972
LightGBM	0.9835	0.9834	0.9835	98.35	0.9970
ExtraTrees	0.9839	0.9839	0.9839	98.39	0.9979
SVC	0.9665	0.9665	0.9665	96.65	0.9921
AdaBoost	0.9473	0.9474	0.9473	94.73	0.9873
Bagging	0.9691	0.9691	0.9691	96.91	0.9904

TABLE III. Base Models' Performance on Dataset 3

Model	Precision	Recall	F1-Score	Accuracy (%)	AUC
RandomForest	0.9656	0.9676	0.9666	96.98	0.9951
GradientBoosting	0.9475	0.9497	0.9486	95.34	0.9893
XGBoost	0.9654	0.9662	0.9658	96.90	0.9951
CatBoost	0.9662	0.9679	0.9671	97.02	0.9952
LightGBM	0.9611	0.9634	0.9622	96.58	0.9944
ExtraTrees	0.9644	0.9664	0.9654	96.86	0.9945
SVC	0.9361	0.9400	0.9380	94.37	0.9829
AdaBoost	0.9300	0.9300	0.9300	93.67	0.9839
Bagging	0.9624	0.9629	0.9626	96.62	0.9891

TABLE IV. Base Models' Performance on Dataset 4

Model	Precision	Recall	F1-Score	Accuracy (%)	AUC
RandomForest	0.9581	0.9578	0.9580	95.81	0.9915
GradientBoosting	0.9303	0.9299	0.9301	93.03	0.9805
XGBoost	0.9574	0.9574	0.9574	95.75	0.9914
CatBoost	0.9575	0.9574	0.9574	95.75	0.9914
LightGBM	0.9510	0.9508	0.9509	95.10	0.9895
ExtraTrees	0.9567	0.9565	0.9566	95.67	0.9913
SVC	0.9139	0.9130	0.9134	91.36	0.9723
AdaBoost	0.9116	0.9118	0.9117	91.18	0.9707
Bagging	0.9521	0.9522	0.9522	95.23	0.9836

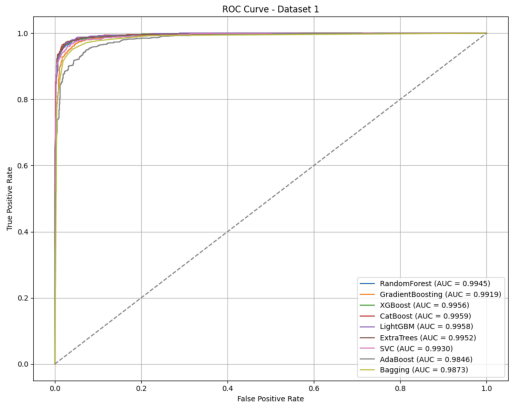


Fig. 2. ROC Curve of base model - Dataset 1.

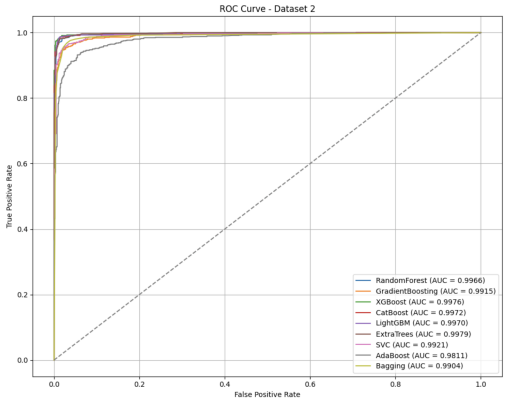


Fig. 3. ROC Curve of base model - Dataset 2.

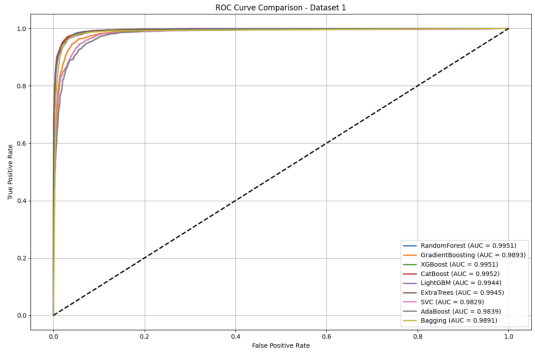


Fig. 4. ROC Curve of base model - Dataset 3.

The results showed that XGBoost and CatBoost consistently outperformed other models across all datasets, especially on the balanced ones, where both achieved perfect or near-perfect scores. Conversely, models like AdaBoost and Gradient Boosting showed performance degradation on imbalanced datasets due to sensitivity to class distribution. Tree-based ensembles such as Random Forest, Extra Trees, and Bagging demonstrated strong stability across all data conditions. SVC generally underperformed, possibly due to its limited scalability in high-dimensional spaces. These findings informed the selection of models used in subsequent ensemble stacking experiments.

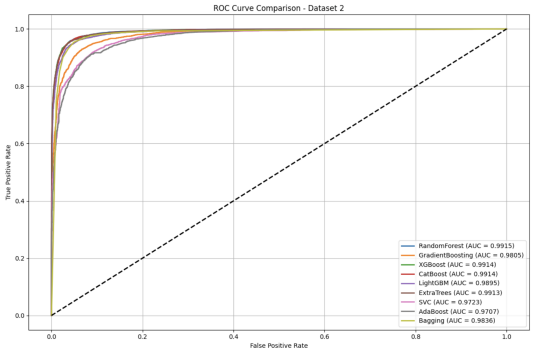


Fig. 5. ROC Curve of base model - Dataset 4.

B. Experiment 2: Performance of Standard Ensemble Stacking Model

This experiment implements a standard ensemble stacking architecture applied consistently across all four datasets. The stacking ensemble integrates tree-based models, including Random Forest and Extra Trees, which are known for handling high-dimensional data and reducing variance, alongside the Bagging classifier to enhance robustness against overfitting. It also incorporates boosting-based classifiers such as Gradient Boosting, XGBoost, LightGBM, and CatBoost, which iteratively improve model accuracy by focusing on difficult samples. Linear and kernel-based models, such as Support Vector Classifier (SVC), are included to capture non-linear patterns, while AdaBoost provides adaptive boosting of weak learners. Logistic Regression is used as the final meta-estimator due to its computational efficiency and ability to aggregate diverse base model outputs effectively.

The same ensemble architecture and base model configuration were used across all datasets to ensure fair comparison. The results, shown in Table V, demonstrate the model's high classification capability across both balanced and imbalanced data distributions. The stacking ensemble achieved high accuracy and consistently strong performance across all datasets. Particularly, Datasets 1 and 2, both balanced and clean, yielded the highest results, with Dataset 2 reaching 98.65% accuracy and an AUC of 0.9983. Dataset 3 and Dataset 4, exhibited slightly lower but still competitive performance. The stacking ensemble demonstrates superior capability in both balanced and imbalanced conditions, confirming its robustness. The high AUC values and consistent classification metrics validate the effectiveness of using heterogeneous learners within a stacking framework. These results justify the ensemble's role as a strong foundation for subsequent optimized stacking improvements.

TABLE V. PERFORMANCE OF STANDARD STACKING ENSEMBLE ACROSS ALL DATASETS

Dataset	Precision	Recall	F1-Score	Accuracy	AUC
Dataset 1	0.97	0.97	0.97	97.38%	0.9962
Dataset 2	0.99	0.99	0.99	98.65%	0.9983
Dataset 3	0.97	0.97	0.97	97.42%	0.9960
Dataset 4	0.96	0.96	0.96	96.34%	0.9935

Fig. 6 and 7 show the ROC curves for Datasets 1 and 2, indicating an ideal classification boundary with near-perfect separation between classes. Fig. 8 and Fig. 9 illustrate the ROC curves for Datasets 3 and 4.

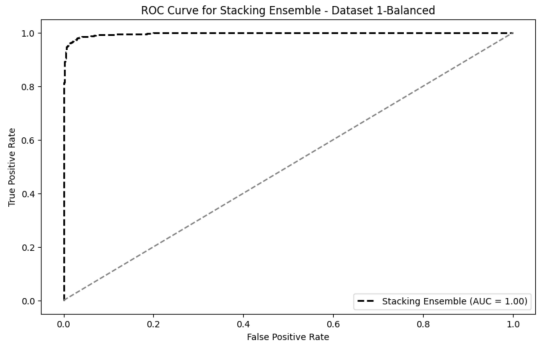


Fig. 6. ROC curve – standard ensemble stacking (Dataset 1).

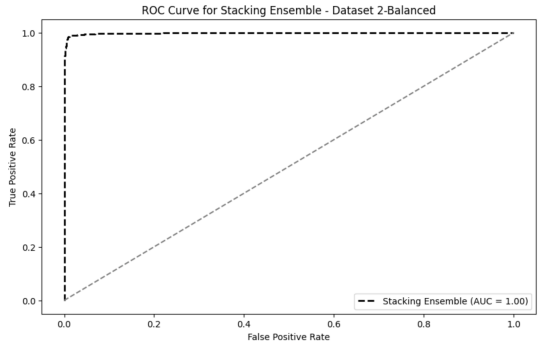


Fig. 7. ROC curve – standard ensemble stacking (Dataset 2).

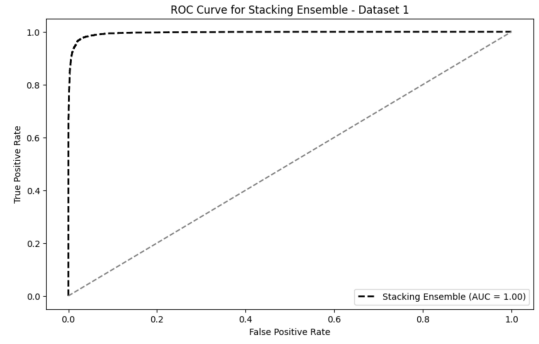


Fig. 8. ROC curve – standard ensemble stacking (Dataset 3).

C. Experiment 3: Optimized Ensemble Stacking Performance

To further enhance classification accuracy and address the limitations of the standard stacking approach, four optimized stacking configurations (Optimized Stacking-1 through Optimized Stacking-4)are implemented and consistently evaluated across four phishing detection datasets. All optimized stacking configurations retained the same architectural structure across all datasets to ensure a fair and consistent comparison. The base models and meta-model remained unchanged, with only hyperparameter adjustments made to accommodate computational constraints, particularly for the larger datasets.

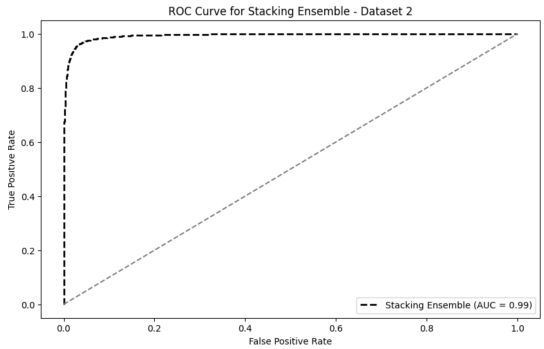


Fig. 9. ROC curve – standard ensemble stacking (Dataset 4).

TABLE VI. PERFORMANCE OF OPTIMIZED STACKING ENSEMBLES ON DATASET 1

Model	Accuracy (%)	Precision	Recall	F1-Score
optimized Stacking-1	100.00	1.00	1.00	1.00
optimized Stacking-2	99.91	1.00	1.00	1.00
optimized Stacking-3	99.78	1.00	1.00	1.00
optimized Stacking-4	99.96	1.00	1.00	1.00

TABLE VII. PERFORMANCE OF OPTIMIZED STACKING ENSEMBLES ON DATASET 2

Model	Accuracy (%)	Precision	Recall	F1-Score
optimized Stacking-1	100.00	1.00	1.00	1.00
optimized Stacking-2	100.00	1.00	1.00	1.00
optimized Stacking-3	100.00	1.00	1.00	1.00
optimized Stacking-4	100.00	1.00	1.00	1.00

TABLE VIII. PERFORMANCE OF OPTIMIZED STACKING ENSEMBLES ON DATASET 3

Model	Accuracy (%)	Precision	Recall	F1-Score
optimized Stacking-1	99.30	0.99	0.99	0.99
optimized Stacking-2	99.71	1.00	1.00	1.00
optimized Stacking-3	99.24	0.99	0.99	0.99
optimized Stacking-4	99.44	0.99	0.99	0.99

TABLE IX. PERFORMANCE OF OPTIMIZED STACKING ENSEMBLES ON DATASET 4

Model	Accuracy (%)	Precision	Recall	F1-Score
optimized Stacking-1	99.27	0.99	0.99	0.99
optimized Stacking-2	99.26	0.99	0.99	0.99
optimized Stacking-3	98.99	0.99	0.99	0.99
optimized Stacking-4	99.09	0.99	0.99	0.99

TABLE X. CROSS-DATASET ACCURACY COMPARISON OF IMPROVED STACKING ENSEMBLES

Model	Dataset 1	Dataset 2	Dataset 3	Dataset 4
optimized Stacking-1	100.00	100.00	99.30	99.27
optimized Stacking-2	99.91	100.00	99.71	99.26
optimized Stacking-3	99.78	100.00	99.24	98.99
optimized Stacking-4	99.96	100.00	99.44	99.09

Tables VI, VII, VIII, and IX demonstrate that all four

optimized stacking ensembles consistently achieved high classification performance across diverse datasets, regardless of balance or size. Optimized Stacking-1, which included a comprehensive set of base classifiers such as gradient boosting, bagging, SVM, and decision trees, reached 100% accuracy on Dataset 1 and Dataset 2 and maintained strong generalization with accuracies of 99.30% and 99.27% on Dataset 3 and Dataset 4, respectively. Optimized Stacking-2 adopted a more streamlined architecture, utilizing only CatBoost, XGBoost, and Random Forest as base models. This approach reduced computational complexity while preserving accuracy, achieving 100% on balanced datasets and the highest recorded accuracy (99.71%) on the large-scale Dataset 3. Optimized Stacking-3 focused on execution efficiency by reducing the ensemble size based on performance impact. While slightly lower in accuracy compared to Ensembles 1 and 2, it still achieved competitive results—99.78% on Dataset 1 and 98.99% on Dataset 4—demonstrating that minimal model diversity can still yield strong outcomes when well-selected. Optimized Stacking-4 further simplified the ensemble configuration to prioritize speed and resource efficiency. Despite this reduction in complexity, it achieved excellent accuracy levels across all datasets, peaking at 99.96% on Dataset 1 and attaining 99.44% on Dataset 3, which confirms its effectiveness for real-time or resource-constrained environments.

As shown in the cross-dataset comparison in Table X, each optimized stacking variant maintained stability and high precision across varied data characteristics. These results collectively validate that the proposed optimized ensemble stacking configurations are not only robust and adaptable but also scalable to different operational constraints and dataset profiles. This makes them highly suitable for deployment in real-world phishing detection systems, where both accuracy and computational efficiency are critical.

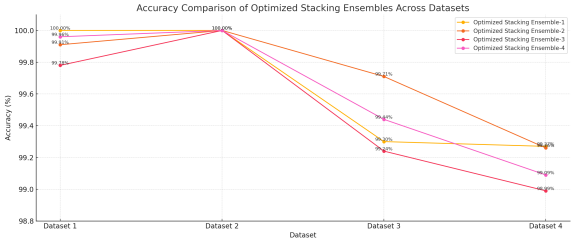


Fig. 10. Comparison of all dataset performance of optimized stacking.

Fig. 11 to Fig. 14 show a comparison of the accuracy of base models,Standard Ensemble Stacking, and improved stacks across datasets.

In addition, Fig. 15 to Fig. 30 illustrate the true and false positives and negatives for the best-performing model on each dataset.

VI. Discussion

We enhance phishing website detection by developing optimized ensemble stacking models and evaluating their performance across four distinct datasets. The Optimized ensemble stacking models outperformed both standard stacking and individual base models due to several key enhancements. First, they utilized optimized feature preprocessing techniques, including SMOTE for class balancing, feature scaling, and label encoding, which allowed for a more accurate representation of phishing patterns. Second, replacing the traditional Logistic Regression meta-model with the more robust CatBoost classifier improved the models' ability to generalize and handle nonlinear relationships and class imbalance. Additionally, reordering and selecting diverse base models enhanced the ensemble's ability to capture complementary learning patterns, resulting in stronger predictions. These improvements also ensured scalability, allowing the models to perform consistently across both small balanced datasets and large imbalanced ones. Finally, the enhanced variants demonstrated significant gains in execution efficiency, maintaining high accuracy while reducing computational time and complexity.

The comparative performance of the optimized ensemble stacking models varies across the four benchmark datasets due to inherent differences in their characteristics. Dataset 1 and Dataset 2 are balanced and relatively clean, with a near 50:50 distribution of phishing and legitimate instances. This balance enables the models to achieve perfect or near-perfect accuracy (approaching 100%), as the absence of significant class imbalance allows the learning process to capture phishing patterns more effectively.

In contrast, Dataset 3 is considerably larger (88,647 instances) and initially imbalanced, necessitating the use of SMOTE to address skewness. Even after balancing, the dataset's scale and the complexity of its features introduce additional variance, which explains the slight reduction in accuracy compared to the smaller, balanced datasets. Similarly, Dataset 4 presents a more challenging structure, with closer ratios between phishing and legitimate cases combined with a higher diversity of complex URL-based features. These factors contribute to minor fluctuations in the achieved results, even when using optimized stacking configurations.

Overall, these findings demonstrate that dataset-specific characteristics—such as class balance, dataset size, and feature richness—play a critical role in shaping model performance. At the same time, the consistently high accuracy achieved across all four datasets highlights the robustness and adaptability of the optimized ensemble stacking framework, which effectively generalizes across both balanced and imbalanced datasets while efficiently handling diverse feature distributions.

A. Dataset Characteristics and their Impact on Model Design

Each of the four datasets used Dataset 1, Dataset 2, Dataset 3, and Dataset 4 presented unique characteristics in terms of size, feature composition, and class distribution, all of which significantly influenced model design choices and performance outcomes. Dataset 1 and Dataset 2 were balanced, with a 50-50 split between legitimate and phishing samples, and exhibited high-quality data with no missing values. Their balanced nature eliminated the need for resampling techniques and allowed direct application of stacking models. Dataset 3 and Dataset 4 were imbalanced, requiring the use of SMOTE to address class distribution skewness. Dataset 3 was particularly large (88,647 records), demanding adjustments to model complexity and training time. Key dataset features influencing stacking performance included URL length, number of special characters, domain-related attributes, redirection behavior, and SSL

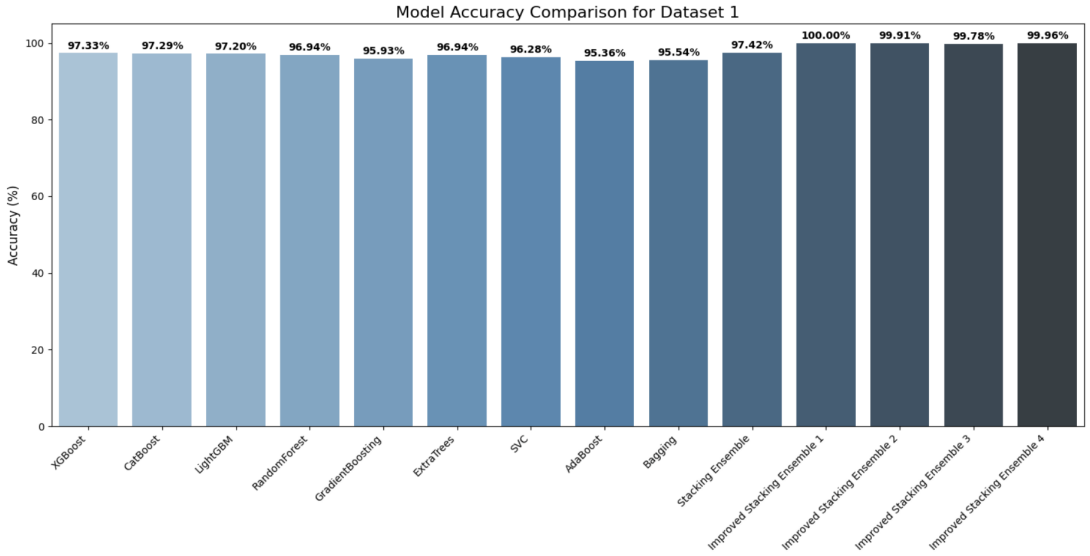


Fig. 11. Accuracy comparison on Dataset 1 for all models.

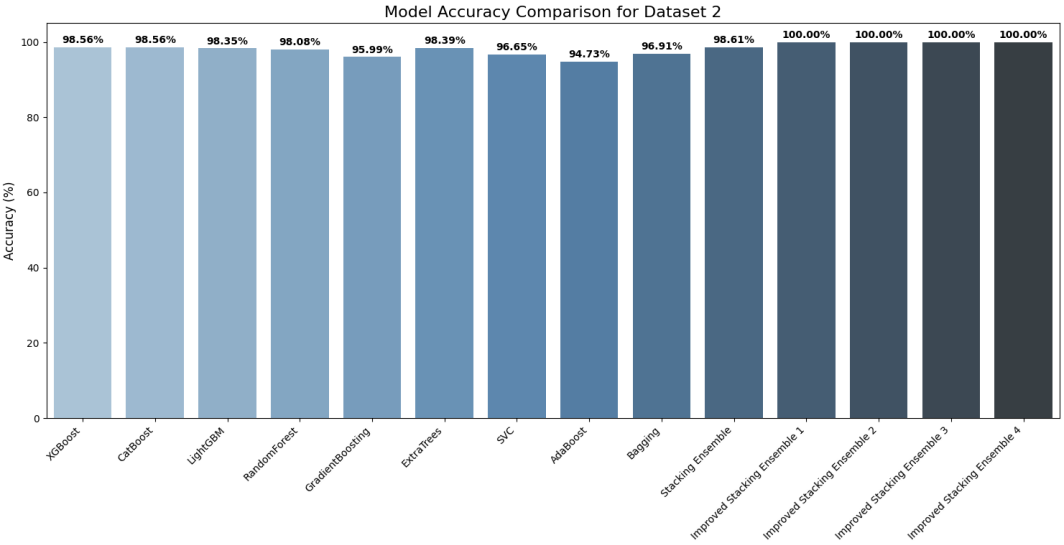


Fig. 12. Accuracy comparison on Dataset 2 for all models.

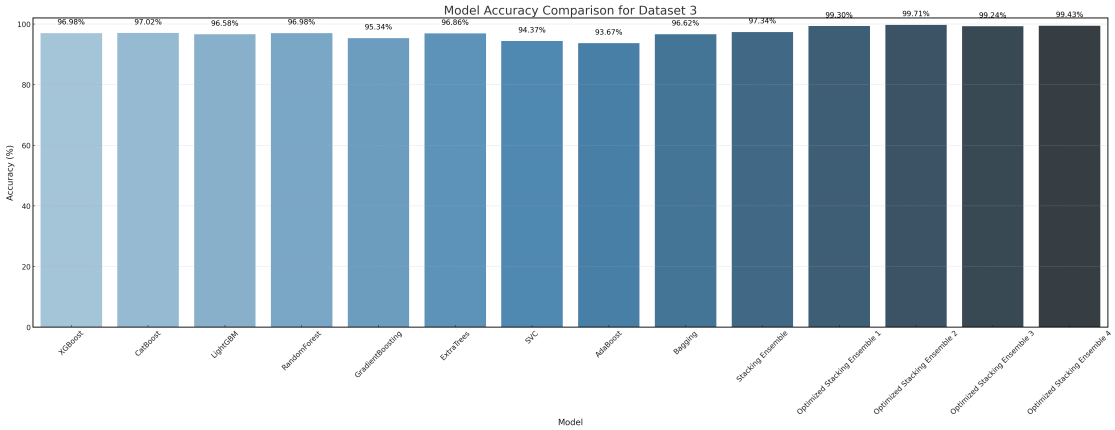


Fig. 13. Accuracy comparison on Dataset 3 for all models.

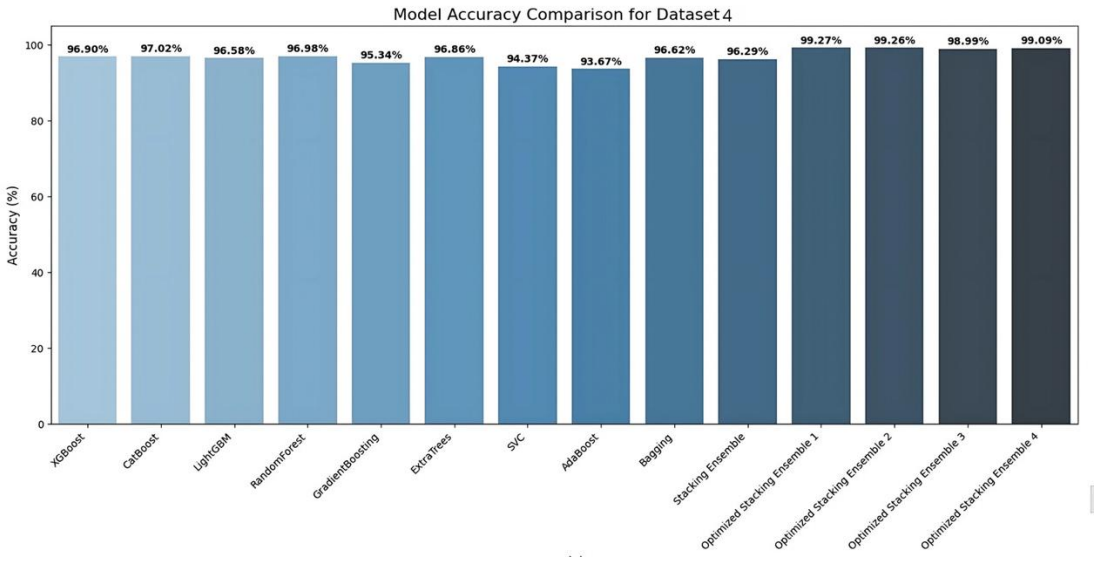


Fig. 14. Accuracy comparison on Dataset 4 for all models.

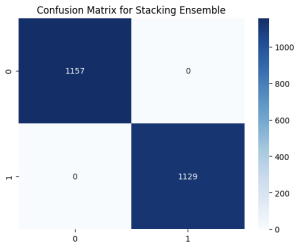


Fig. 15. Optimized ensemble stacking-1 (Dataset 1).

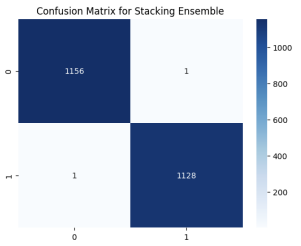


Fig. 16. Optimized ensemble stacking-2 (Dataset 1).

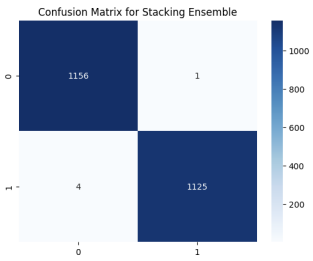


Fig. 17. Optimized ensemble stacking-3 (Dataset 1).

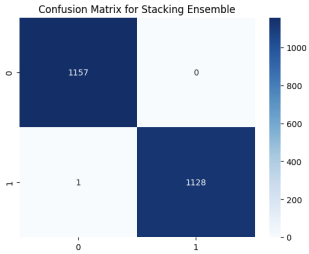


Fig. 18. Optimized ensemble stacking-4 (Dataset 1).

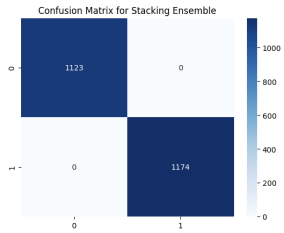


Fig. 19. Optimized ensemble stacking-1 (Dataset 2).

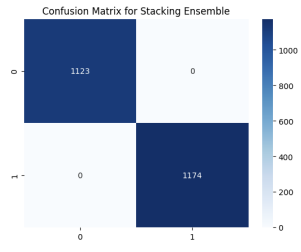


Fig. 20. Optimized ensemble stacking-2 (Dataset 2).

certificate indicators. These were effectively captured through hybrid feature engineering strategies.

To adapt to each dataset's scale and quality:

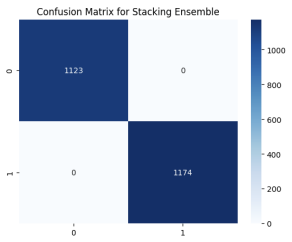


Fig. 21. Optimized ensemble stacking-3 (Dataset 2).

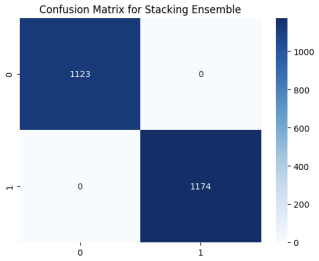


Fig. 22. Optimized ensemble stacking-4 (Dataset 2).

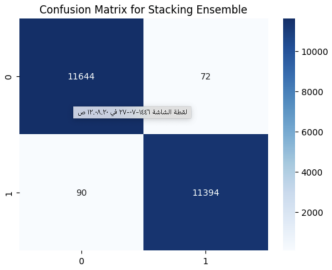


Fig. 23. Optimized ensemble stacking-1 (Dataset 3).

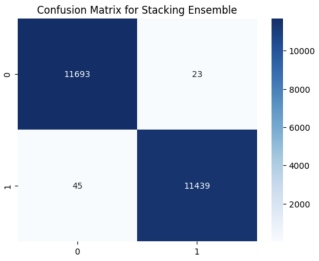


Fig. 24. Optimized ensemble stacking-2 (Dataset 3).

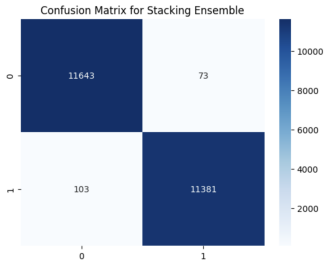


Fig. 25. Optimized ensemble stacking-3 (Dataset 3).

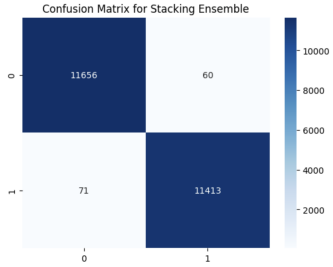


Fig. 26. Optimized ensemble stacking-4 (Dataset 3).

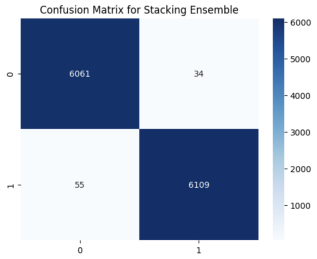


Fig. 27. Optimized ensemble stacking-1 (Dataset 4).

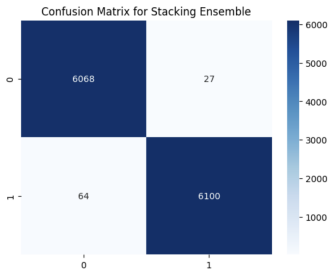


Fig. 28. Optimized ensemble stacking-2 (Dataset 4).

Feature encoding and scaling were uniformly applied. SMOTE was selectively used for unbalanced datasets. Cross-validation (StratifiedKFold) ensured consistent generalization across all experiments. Ultimately, the characteristics of each dataset informed key design decisions in the stacking architecture. While the same base and meta-model configurations were used across all datasets to ensure comparability, larger and imbalanced datasets necessitated specific adjustments—such as the application of SMOTE, reduced model iterations, and fewer cross-validation folds—to manage computational complexity without sacrificing accuracy. preprocessing, standardization, and encoding strategies were uniformly applied. These adaptations highlight how dataset size, balance, and structure influenced model pruning strategies and the depth of optimization required to achieve efficient and high-performing ensemble detection systems.

B. Performance Comparison of the Proposed Approach with Previous Studies

To objectively assess the effectiveness of the proposed optimized ensemble stacking model, it is essential to compare its performance with existing approaches reported in the literature.

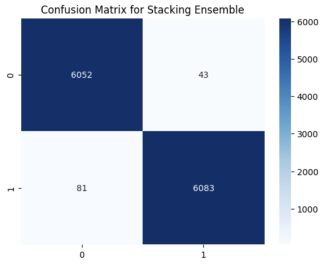


Fig. 29. Optimized ensemble stacking-3 (Dataset 4).

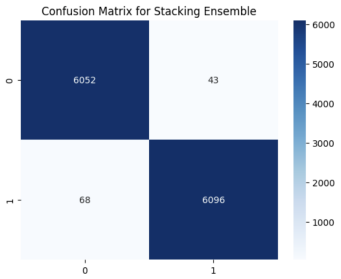


Fig. 30. Optimized ensemble stacking-4 (Dataset 4).

Several recent studies have explored various ensemble learning techniques to enhance phishing website detection, including optimized stacking, such as Genetic Algorithm (GA), to optimize the parameters of various ensemble classifiers, Diverse Base Learners, Hyperparameter Tuning, multilayer-stacked models, and boosting-based ensembles. This section presents a comparative analysis of classification accuracy across these studies, focusing on the same or similar benchmark datasets. A comparative summary of classification accuracy reported in these studies, focusing on similar benchmark datasets, is provided in Table XI.

TABLE XI. COMPARISON WITH RELATED STUDIES OF THE PROPOSED METHOD

Paper	Classifiers	Accuracy (%)
[5]	optimized stacking	D3:97.3%
[9]	Multilayer-Stacking	D3:96.79%, D4:98.43%
[15]	Boosting-based multi-layer stacked ensemble model	D4:96.16%, D3:98.95%
[4]	Stacking ensemble classifier	D1:98.20, D3:97.48%

Our enhanced ensemble stacking architecture clearly demonstrates advantages over existing methods. For example, [5] achieved 97.3% accuracy on Dataset 3 using an optimized stacking approach, while [9] reported 96.79% on Dataset 3 and 98.43% on Dataset 4 with multilayer stacking. Similarly, boosting-based multi-layer ensembles [15] obtained 98.95% on Dataset 3 and 96.16% on Dataset 4. In contrast, our optimized stacking variants consistently exceed 99% accuracy on the same datasets, with Optimized Stacking-2 reaching 99.71% on Dataset 3 and 99.28% on Dataset 4. Furthermore, while [4] reported 97.48% accuracy on Dataset 3, our method achieved over 99% under the same conditions. These comparisons highlight that our approach not only achieves higher accuracy, but also exhibits greater robustness when applied to imbalanced datasets (Datasets 3 and 4), where many prior methods suffered performance drops. By integrating CatBoost as a meta-learner, employing diverse base classifiers, and applying robust validation strategies, our model advances the state-of-the-art in phishing website detection.

C. Limitations

While the optimized ensemble stacking models achieved state-of-the-art performance across multiple datasets, several limitations should be acknowledged. First, training complex ensembles with multiple strong base learners and a CatBoost meta-model incurs significant computational cost and longer execution times, particularly for very large datasets. Second, achieving 100% accuracy on smaller balanced datasets may indicate a potential risk of overfitting, underscoring the need for careful validation on real-world, unseen data. Third, phishing tactics continuously evolve, meaning that static models trained on historical datasets may degrade in performance over time. Therefore, periodic retraining and the incorporation of adaptive or online learning mechanisms are essential to maintain robustness.

Finally, while SMOTE and preprocessing strategies helped address class imbalance, synthetic oversampling may not fully capture the diversity of real phishing examples. Future work should validate the models on live network traffic and diverse sources to ensure practical applicability.

VII. CONCLUSIONS

This study demonstrated the effectiveness of optimized ensemble stacking models in enhancing phishing website detection across datasets with diverse characteristics. The proposed framework consistently achieved superior results, including 100% accuracy on the two balanced datasets (Datasets 1 and 2) and above 99% accuracy on the larger and imbalanced datasets (Datasets 3 and 4). These outcomes confirm the robustness and scalability of the approach under varying data conditions. Among the four optimized variants, Optimized Stacking-2 and Optimized Stacking-4 provided the best trade-off between detection accuracy and computational efficiency, making them well-suited for real-world deployment.

The key contributions of this work include: systematic optimization of diverse base learners within the stacking framework, replacement of traditional Logistic Regression with CatBoost as the meta-classifier to improve generalization and handle imbalance, and comprehensive validation across multiple datasets to ensure reliability and robustness. These advances distinguish the proposed method from prior approaches that often relied on fixed classifiers, limited datasets, or less rigorous optimization.

From a practical standpoint, the findings highlight that phishing detection systems can achieve both high accuracy and efficiency by adopting optimized ensemble stacking. The integration of explainability tools such as SHAP further enhances usability by providing interpretable insights for security analysts, enabling informed decision-making in operational environments.

Future work will focus on extending this research by validating the framework on live phishing traffic, evaluating inference latency for real-time deployment, and exploring semi-supervised or adversarial learning techniques to counter

evolving zero-day attacks. Additionally, lightweight versions of the model will be developed to improve adaptability in resource-constrained environments, such as IoT and mobile devices.

In conclusion, this research establishes optimized ensemble stacking as a highly effective and practical solution for phishing website detection, offering both state-of-the-art accuracy and strong adaptability to dynamic cybersecurity challenges.

REFERENCES

[1] S. Aslam, H. Aslam, A. Manzoor, H. Chen, and A. Rasool, "Antiphishstack: Lstm-based stacked generalization model for optimized phishing url detection," *Symmetry*, vol. 16, no. 2, p. 248, 2024.

[2] F. S. Bidabadi and S. Wang, "A new weighted ensemble model for phishing detection based on feature selection," *arXiv preprint arXiv:2212.11125*, 2022.

[3] P. Meena, P. Singla, and P. Ranjan, "Enhanced phishing url detection through stacked machine learning model," in *2024 International Conference on Intelligent Systems for Cybersecurity (ISCS)*. IEEE, 2024, pp. 1–6.

[4] A. Newaz, F. S. Haq, and N. Ahmed, "A sophisticated framework for the accurate detection of phishing websites," *arXiv preprint arXiv:2403.09735*, 2024.

[5] M. Al-Sarem, F. Saeed, Z. G. Al-Mekhlafi, B. A. Mohammed, T. Al-Hadhrami, M. T. Alshammari, A. Alreshidi, and T. S. Alshammari, "An optimized stacking ensemble model for phishing websites detection," *Electronics*, vol. 10, no. 11, p. 1285, 2021.

[6] S. S. M. M. Rahman, T. Islam, and M. I. Jabiullah, "Phishstack: evaluation of stacked generalization in phishing urls detection," *Procedia Computer Science*, vol. 167, pp. 2410–2418, 2020.

[7] A. F. Nugraha, R. F. A. Aziza, and Y. Pristyanto, "Penerapan metode stacking dan random forest untuk meningkatkan kinerja klasifikasi pada proses deteksi web phishing," *Jurnal Infomedia: Teknik Informatika, Multimedia, dan Jaringan*, vol. 7, no. 1, pp. 39–44, 2022.

[8] D. W. Kiseki, V. Havyarimana, L. Zabagunda, W. I. Wail, T. Niyonsaba *et al.*, "Artificial intelligence in cybersecurity to detect phishing," *Journal of Computer and Communications*, vol. 12, no. 12, pp. 91–115, 2024.

[9] L. R. Kalabarige, R. S. Rao, A. Abraham, and L. A. Gabralla, "Multilayer stacked ensemble learning model to detect phishing websites," *Ieee Access*, vol. 10, pp. 79 543–79 552, 2022.

[10] H. A. Wabi, J. A. Ojeniyi, I. Idris, and S. O. Subairu, "Stack ensemble model for detection of phishing website," in *2024 International Conference on Science, Engineering and Business for Driving Sustainable Development Goals (SEB4SDG)*. IEEE, 2024, pp. 1–6.

[11] Z. G. Al-Mekhlafi, B. A. Mohammed, M. Al-Sarem, F. Saeed, T. A. Hadhrami, M. T. Alshammari, A. Alreshidi, and T. S. Alshammari, "Phishing websites detection by using optimized stacking ensemble model," *Comput. Syst. Sci. Eng.*, vol. 41, pp. 109–125, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:240426096

[12] B. A. Mohammed and Z. G. Al-Mekhlafi, "Optimized stacking ensemble model to detect phishing websites," in *International Conference on Advances in Cybersecurity*, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:244846728

[13] S. Giri and S. Banerjee, "Ensemble learning approach for phishing website detection using an optimal greedy stacking model," *Journal of The Institution of Engineers (India): Series B*, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:272639107

[14] M. K. Pandey, M. K. Singh, S. Pal, and B. B. Tiwari, "Prediction of phishing websites using stacked ensemble method and hybrid features selection method," *SN Computer Science*, vol. 3, pp. 1–11, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:252498524

[15] L. R. Kalabarige, R. S. Rao, A. R. Pais, and L. A. Gabralla, "A boosting-based hybrid feature selection and multi-layer stacked ensemble learning model to detect phishing websites," *IEEE Access*, vol. 11, pp. 71 180–71 193, 2023.

[16] N. Belsare, P. Sonaje, A. Ahmed, V. Gugale, S. Barve, and D. Chikmurge, "Enhancing phishing website detection using stacking ensemble techniques," *2024 3rd International Conference for Advancement in Technology (ICONAT)*, pp. 1–9, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:274641655

[17] R. Soleymanzadeh, M. Aljasim, M. W. Qadeer, and R. Kashef, "Cyberattack and fraud detection using ensemble stacking," *AI*, vol. 3, no. 1, pp. 22–36, 2022.

[18] M. A. Alsharaiah, A. A. Abu-Shareha, M. Abualhaj, L. H. Baniata, O. Adwan, A. Al-Saaidah, and M. Oraiqat, "A new phishing-website detection framework using ensemble classification and clustering," 2023.

[19] Y. Wei and Y. Sekiya, "Sufficiency of ensemble machine learning methods for phishing websites detection," *IEEE Access*, vol. 10, pp. 124 103–124 113, 2022.

[20] N. Lawrance, M.-A. Guerry, and G. Petrides, "Cost-sensitive stacking: an empirical evaluation," *arXiv preprint arXiv:2301.01748*, 2023.

[21] G. Dharmaraju, T. N. Kumar, P. P. Mohan, R. R. Pbv, and A. Lakshmanarao, "Phishing website detection through ensemble machine learning techniques," in *2024 2nd International Conference on Computer, Communication and Control (IC4)*. IEEE, 2024, pp. 1–5.

[22] Z. Ghaleb Al-Mekhlafi, B. Abdulkarem Mohammed, M. Al-Sarem, F. Saeed, T. Al-Hadhrami, M. T. Alshammari, A. Alreshidi, and T. Sarheed Alshammari, "Phishing websites detection by using optimized stacking ensemble model," *Computer Systems Science and Engineering*, vol. 41, no. 1, pp. 109–125, 2022.

[23] Y. Li, Z. Yang, X. Chen, H. Yuan, and W. Liu, "A stacking model using url and html features for phishing webpage detection," *Future Generation Computer Systems*, vol. 94, pp. 27–39, 2019.

[24] H. Baliyan and A. R. Prasath, "Enhancing phishing website detection using ensemble machine learning models," in *2024 OPJU International Technology Conference (OTCON) on Smart Computing for Innovation and Advancement in Industry 4.0*. IEEE, 2024, pp. 1–8.

[25] M. S. I. Ovi, M. H. Rahman, and M. A. Hossain, "Phishguard: A multilayered ensemble model for optimal phishing website detection," *arXiv preprint arXiv:2409.19825*, 2024.

[26] A. Hannousse, "Web page phishing detection [data set]," https://doi.org/10.17632/c2gw7fy2j4.1, 2020, mendeley Data.

[27] M. K. Chaurasia, "Web page phishing detection [data set]," https://www.kaggle.com/datasets/manishkc06/web-page-phishing-detection, 2021, kaggle.

[28] G. Vrbancic, "Phishing websites dataset," https://data.mendeley.com/datasets/72ptz43s9v/1, 2020, mendeley Data, vol. 1.

[29] J. Kolla, S. Praneeth, M. S. Baig, and G. reddy Karri, "A comparison study of machine learning techniques for phishing detection," *Journal Of Business And Information Systems (e-ISSN: 2685-2543)*, vol. 4, no. 1, pp. 21–33, 2022.

[30] K. Omari, "Comparative study of machine learning algorithms for phishing website detection," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 9, 2023.

[31] M. R. Chinguwo and R. Dhanalakshmi, "Detecting cloud based phishing attacks using stacking ensemble machine learning technique," *International Journal for Research in Applied Science & Engineering Technology (IJRASET). ISSN*, pp. 2321–9653, 2023.

[32] B. Deekshitha, C. Aswitha, C. S. Sundar, and A. K. Deepthi, "Url based phishing website detection by using gradient and catboost algorithms," *Int. J. Res. Appl. Sci. Eng. Technol*, vol. 10, no. 6, pp. 3717–3722, 2022.

[33] M. Khatun, A. Mozumder, N. Polash, M. R. Hasan, K. Ahammad, and M. S. Shaiham, "An approach to detect phishing websites with features selection method and ensemble learning," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 13, no. 8, 2022.

[34] M. Z. Hassan, M. H. Kabir, and M. A. Hamja, "Phishing website identification: Unleashing the potential of machine learning and stacking ensembles techniques," *Journal of Science and Technology*, vol. 122, p. 129, 2024.

[35] X. Zhou and R. M. Verma, "Phishing sites detection from a web developer's perspective using machine learning." in *HICSS*, 2020, pp. 1–10.

[36] scikit-learn developers, *scikit-learn: Machine Learning in Python*, 2024, version 1.7.0. [Online]. Available: https://scikit-learn.org

[37] XGBoost Developers, *XGBoost: Scalable, Portable and Distributed Gradient Boosting*, 2024, version 3.1.0. [Online]. Available: https://xgboost.readthedocs.io

[38] LightGBM Developers, *LightGBM: A Fast, Distributed, High Performance Gradient Boosting (GBDT, GBRT, GBM or MART) Framework*, 2024, version 4.3.0. [Online]. Available: https://lightgbm.readthedocs.io

[39] CatBoost Developers, *CatBoost: Gradient Boosting with Categorical Features Support*, 2024, version 1.2.3. [Online]. Available: https://catboost.ai

[40] Hunter, J. D. and matplotlib development team, *matplotlib: Visualization with Python*, 2024, version 3.8.4. [Online]. Available: https://matplotlib.org

[41] Waskom, M. L. and seaborn development team, *seaborn: Statistical Data Visualization*, 2024, version 0.13.2. [Online]. Available: https://seaborn.pydata.org

[42] pandas development team, *pandas: Powerful Python Data Analysis Toolkit*, 2024, version 2.2.2. [Online]. Available: https://pandas.pydata.org

[43] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith *et al.*, "Array programming with numpy," *Nature*, vol. 585, no. 7825, pp. 357–362, 2020.

Grey Clustering Algorithm for Urban Air Quality Classification: A Case Study in Lima, Peru

Alexi Delgado, Katherine Paredes Guerrero, Anderson Carrillo
Environmental Engineering Program, Universidad de Ciencias y Humanidades, Lima, Peru

***Abstract*—This study introduces a grey clustering algorithm based on the Central Triangular Whitenization Weight Function (CTWF), designed to classify urban air quality under conditions of limited or uncertain data. Based on Grey Systems Theory (GST), the proposed algorithm facilitates structured multi-criteria assessments using sparse or irregular datasets—a condition frequently encountered in urban environmental monitoring. The algorithm stands out for its low computational complexity, interpretability, and ability to integrate multiple pollutants into a single qualitative classification, making it particularly suitable for smart city applications and real-time decision support systems. To evaluate its performance, the grey clustering algorithm (CTWF) was applied to a case study in Northern Lima, Peru, covering eight semesters between 2011 and 2019 and including four key pollutants: PM_{10}, SO_2, NO_2, and CO. Although all periods were classified as "Good" under national standards, the disaggregated analysis revealed PM_{10} as the most persistent concern, while CO levels remained consistently low, and SO_2 and NO_2 showed moderate fluctuations. These findings validate the algorithm's capacity to extract pollutant-specific insights and spatiotemporal trends even in data-scarce environments. Future enhancements may include meteorological integration, broader pollutant sets (e.g., $PM_{2.5}$, ozone), and satellite data to extend forecasting capabilities and spatial resolution.**

***Keywords*—Grey clustering algorithm; air quality classification; grey systems theory; urban air pollution**

I. Introduction

Urban air pollution remains a critical and escalating environmental concern, particularly in rapidly growing cities in the global South, where population growth often outpaces infrastructure development. Its detrimental effects on respiratory health, mortality, and environmental sustainability are well documented, especially in low- and middle-income countries [1], [2], [3]. In Latin America, cities such as Lima, São Paulo, and Bogotá routinely report air pollutant concentrations that exceed both national regulations and international safety guidelines [4].

In the case of Lima, Peru, the second largest city in South America, unregulated urban sprawl, high vehicular density, and industrial activity have all contributed to significant air quality degradation [5], [6]. This issue is especially pronounced in northern districts such as Comas, Los Olivos, and Carabayllo, where socioeconomically vulnerable populations are exposed to elevated pollution levels but lack access to comprehensive monitoring systems [7]. Despite growing public concern, air quality assessments remain constrained by fragmented datasets, limited monitoring infrastructure, and the absence of tools designed to function under data uncertainty [8], [9].

These challenges have prompted the need for alternative modeling approaches that are capable of generating reliable insights from minimal or limited data. In this context, Grey Systems Theory (GST), introduced by Deng in the early 1980s, offers a mathematical framework tailored for decision-making under conditions of limited or uncertain information [10], [11]. Unlike traditional statistical models, which rely on consistent and extensive datasets, GST is designed to function effectively even when sample sizes are small or measurements are incomplete, conditions commonly found in urban environmental assessments in Latin America [12], [13].

The flexibility and scalability of GST have led to its increasing application in environmental engineering. Studies have used GST to analyze water quality [14], assess environmental impacts [15], or uncertain environments [16]. Among its techniques, grey clustering has gained attention for its capacity to classify environmental conditions using minimal but representative datasets [17]. These models assign qualitative categories (for example, "Good", "Unhealthy") to pollution levels by mapping quantitative indicators through triangular membership functions, allowing intuitive yet rigorous environmental assessments [18].

In recent years, the integration of GST with entropy weighting, fuzzy logic, and multi-criteria decision making (MCDM) has further expanded its utility, particularly in complex scenarios with multiple and overlapping pollution sources [19], [20], [21]. Such hybrid models offer enhanced classification accuracy and robustness, particularly in urban areas where pollution is influenced by dynamic factors such as traffic, industry, and meteorology [22].

Given these advantages, this study develops and applies an algorithm based on the Central Triangular Whitenization Weight Function (CTWF), a specific grey clustering model, to classify historical air quality conditions in Northern Lima over an eight-semesters period (2011 to 2019). Drawing on publicly available concentration data for four primary pollutants (PM_{10}, SO_2, NO_2, and CO), the grey clustering algorithm (CTWF) enables a structured multi-criteria classification aligned with national Environmental Quality Standards (EQS). This methodological framework is particularly valuable in urban settings like Lima, where technical and governance constraints limit the use of conventional monitoring infrastructure.

Therefore, this work pursues two main objectives:

- To develop a grey clustering algorithm (CTWF) to classify urban air quality; and
- To evaluate the performance of the algorithm through a case study in Lima, Peru.

Overall, this study contributes to applied environmental engineering by demonstrating how intelligent clustering tools

like CTWF can support monitoring, analysis, and decision-making in urban environments where conventional systems fall short. Its methodological simplicity and replicability also position it as a practical tool for use in similar data-constrained urban contexts globally.

This study is organized as follows: Section I provides the introduction. In Section II, the grey clustering algorithm is presented. In Section III, the case study is developed and in Section IV the results are discussed. Finally, in Section V the conclusions are presented.

II. Grey Clustering Algorithm

In this section the grey clustering algorithm for air quality classification, based on CTWF, is developed.

A. Definition

First, to develop the grey clustering algorithm (CTWF), the following components are defined:

- Study object m study objects are defined O_1, O_2, O_3, ..., O_m
- Criteria n criteria are defined C_1, C_2, C_3, ..., C_n
- Grey classes s grey classes are defined S_1, S_2, S_3, ..., S_s

B. Algorithm Structure and Computational Steps

The steps of the algorithm proposed in this work are detailed below [14]:

1) Step 1 – Center-point determination: The center-points of the s grey classes are determined, the sequence of the center-points is: $\mu_1, \mu_2, \mu_3, ..., \mu_s$.

2) Step 2 – Data normalization: The center-points of the grey classes, for each criterion, are normalized using Eq. (1):

$$\lambda_s = \frac{\mu_s}{\bar{y}_s} \tag{1}$$

where, $\bar{y}_s$ is the mean of the center-points of grey classes for each criterion. Then, the sample data (X_{ij}) are normalized using Eq. (2):

$$x_{ij} = \frac{X_{ij}}{\bar{y}_s} \tag{2}$$

3) Step 3 – Membership function (CTWF): Each normalized value is transformed into a triangular membership function centered at a reference class midpoint, as shown in Fig. 1 and Eq. (3) to Eq. (5):

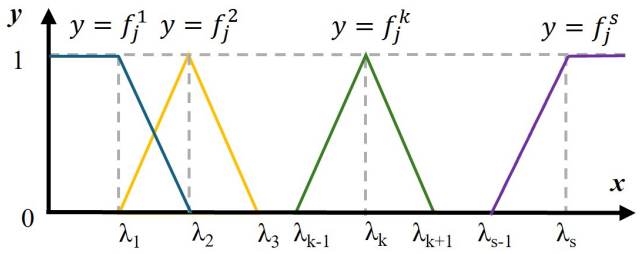


Fig. 1. CTWF graphical representation.

$$f_j^1(x_{ij}) = \begin{cases} 0, & x \notin [0, \lambda_2] \\ 1, & x \in [0, \lambda_1] \\ \dfrac{\lambda_2 - x}{\lambda_2 - \lambda_1}, & x \in [\lambda_1, \lambda_2] \end{cases} \tag{3}$$

$$f_j^k(x_{ij}) = \begin{cases} 0, & x \notin [\lambda_{k-1}, \lambda_{k+1}] \\ \dfrac{x - \lambda_{k-1}}{\lambda_k - \lambda_{k-1}}, & x \in [\lambda_{k-1}, \lambda_k] \\ \dfrac{\lambda_{k+1} - x}{\lambda_{k+1} - \lambda_k}, & x \in [\lambda_k, \lambda_{k+1}] \end{cases} \tag{4}$$

$$f_j^s(x_{ij}) = \begin{cases} 0, & x \notin [\lambda_{s-1}, +\infty) \\ \dfrac{x - \lambda_{s-1}}{\lambda_s - \lambda_{s-1}}, & x \in [\lambda_{s-1}, \lambda_s] \\ 1, & x \in [\lambda_s, +\infty) \end{cases} \tag{5}$$

4) Step 4 – Criteria weight calculation: Relative weights are assigned to each criterion using Eq. (6) [23].

$$\eta_j^k = \frac{\dfrac{1}{\lambda_j^k}}{\sum_{j=1}^{m} \dfrac{1}{\lambda_j^k}} \tag{6}$$

5) Step 5 – Grey clustering coefficient computation: Clustering coefficients are calculated for each study object and grey class, integrating both the CTWF value and the corresponding weight of each criterion by Eq. (7):

$$\sigma_j^k = \sum_{j=1}^{m} f_j^k(x_{ij}) \cdot \eta_j^k \tag{7}$$

6) Step 6 – Final classification: Each study object is assigned to the grey class with the highest clustering coefficient using Eq. (8):

$$\sigma_j^{k^*} = \max_{1 \le k \le s} \{\sigma_j^k\} \tag{8}$$

III. Case Study in Lima, Peru

Lima, the capital of Peru, is a coastal mega-city of over ten million inhabitants and is among the most air-polluted cities in South America [24]. Characterized by high vehicular density, poor atmospheric dispersion, and unregulated urban growth, Lima's air pollution is driven largely by emissions from the transportation sector and informal waste burning.

The northern districts of Lima, including Comas, Carabayllo, Los Olivos, San Martín de Porres, and Puente Piedra, exemplify these challenges. These areas combine high residential density with commercial corridors and are under constant urban expansion. Meteorological conditions such as thermal inversion and limited wind flow further exacerbate pollutant accumulation [25]. Urban infrastructure and environmental monitoring are weaker in these districts, which are often socioeconomically marginalized and environmentally underserved [26]. The evaluation of this air quality study was carried out in metropolitan Lima, specifically in northern Lima, in Los Olivos district, as shown in Fig. 2.

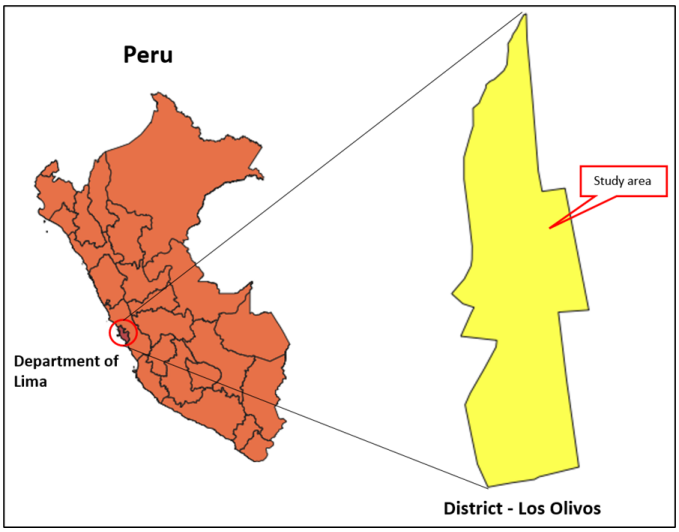


Fig. 2. Location of the study area.

A. Application of the Definition

The components for the case study were defined as follows:

1) Study object: Eight study objects were defined according to the semesters studied from 2001 to 2019 years. The study objects are presented in Table I .

2) Criteria: Four criteria were studied. The definition of these criteria was based on air pollution parameters, considering the following main pollutants: PM_{10}, SO_2, NO_2, and CO. The collected data (in $\mu g/m^3$) for each study object are presented in Table I [27].

3) Grey classes: Four grey classes were defined, based on Peru's Environmental Quality Standards (EQS) for PM_{10}, SO_2, NO_2, and CO parameters, as shown in Table II [28].

B. Application of Algorithm steps

The steps of the algorithm proposed in this work were applied to the case study as shown below:

TABLE I. Study Objects, Criteria, and Monitoring Data

Study Object	Semester	PM_{10} (C1)	SO_2 (C2)	NO_2 (C3)	CO (C4)
O1	2011-1	85.6	13.5	3.8	3.1
O2	2012-1	134.2	11.8	3.7	2.6
O3	2013-1	117.8	13.0	156.0	1809
O4	2014-1	143.2	12.15	78.35	600
O5	2015-1	165.4	13.0	45.46	1599
O6	2016-1	100.56	13.0	66.81	600
O7	2018-1	99.12	12.15	8.75	652
O8	2019-1	95.48	13.0	3.33	600

TABLE II. Grey Classes of the Case Study

Air Quality Level	Range ($\mu g/m^3$)	Grey Class
PM_{10} (C1)		
Good	0–75	S1
Moderate	76–150	S2
Unhealthy for sensitive groups	151–250	S3
Unhealthy	251–350	S4
SO_2 (C2)		
Good	0–20	S1
Moderate	21–80	S2
Unhealthy for sensitive groups	81–500	S3
Unhealthy	501–920	S4
NO_2 (C3)		
Good	0–100	S1
Moderate	101–200	S2
Unhealthy for sensitive groups	201–300	S3
Unhealthy	301–400	S4
CO (C4)		
Good	0–5000	S1
Moderate	5001–10000	S2
Unhealthy for sensitive groups	10001–15000	S3
Unhealthy	15001–20000	S4

1) Step 1: From Table II, the center-points of the four grey classes, in each criterion, were determined. The results are presented in Table III.

TABLE III. Center-Points of the Grey Classes

Class	C_1 (PM_{10})	C_2 (SO_2)	C_3 (NO_2)	C_4 (CO)
S1	37.5	10.0	50.0	2500.0
S2	113.0	50.5	150.5	7500.5
S3	200.5	290.5	250.5	12500.5
S4	300.5	710.5	350.5	17500.5

2) Step 2: From Table III, the center-points of the grey classes, for each criterion, were normalized using Eq. (1). The results are presented in Table IV.

TABLE IV. Normalized Values for Grey Classes

Criterion	λ_1	λ_2	λ_3	λ_4
C_1	0.230	0.694	1.231	1.845
C_2	0.038	0.190	1.095	2.677
C_3	0.250	0.751	1.250	1.749
C_4	0.250	0.750	1.250	1.750

Then, from Table I , the sample values, for each criterion, were normalized using Eq. (2). The results are presented in Table V.

3) Step 3: Each normalized value was transformed into a triangular membership function centered at a reference class

TABLE V. NORMALIZED VALUES FOR SAMPLE VALUES

Study Object	C_1	C_2	C_3	C_4
O1	0.526	0.051	0.019	0.000
O2	0.824	0.044	0.018	0.000
O3	0.723	0.049	0.779	0.181
O4	0.879	0.046	0.391	0.060
O5	1.016	0.049	0.227	0.160
O6	0.617	0.049	0.333	0.060
O7	0.609	0.046	0.044	0.065
O8	0.586	0.049	0.017	0.060

midpoint by Eq. (3) to Eq. (5). The equations were then synthesized and adapted to the four grey classes, resulting in the following base Eq. (8) to Eq. (11).

$$f_j^1(x_{ij}) = \begin{cases} 0, & x \notin [0, \lambda_2] \\ 1, & x \in [0, \lambda_1] \\ \dfrac{\lambda_2 - x}{\lambda_2 - \lambda_1}, & x \in [\lambda_1, \lambda_2] \end{cases} \tag{8}$$

$$f_j^2(x_{ij}) = \begin{cases} 0, & x \notin [\lambda_1, \lambda_3] \\ \dfrac{x - \lambda_1}{\lambda_2 - \lambda_1}, & x \in [\lambda_1, \lambda_2) \\ \dfrac{\lambda_3 - x}{\lambda_3 - \lambda_2}, & x \in [\lambda_2, \lambda_3] \end{cases} \tag{9}$$

$$f_j^3(x_{ij}) = \begin{cases} 0, & x \notin [\lambda_2, \lambda_4] \\ \dfrac{x - \lambda_2}{\lambda_3 - \lambda_2}, & x \in [\lambda_2, \lambda_3) \\ \dfrac{\lambda_4 - x}{\lambda_4 - \lambda_3}, & x \in [\lambda_3, \lambda_4] \end{cases} \tag{10}$$

$$f_j^4(x_{ij}) = \begin{cases} 0, & x \notin [\lambda_3, +\infty) \\ \dfrac{x - \lambda_3}{\lambda_4 - \lambda_3}, & x \in [\lambda_3, \lambda_4) \\ 1, & x \in [\lambda_4, +\infty) \end{cases} \tag{11}$$

The values from Table IV and Table V were then substituted into Eq. (8) to Eq. (11). As an example, the results of O_1 study object are presented in Table VI.

TABLE VI. CTWF VALUES FOR STUDY OBJECT O_1

Function	C_1	C_2	C_3	C_4
f_j^1	0.362	0.914	1.000	1.000
f_j^2	0.638	0.086	0.000	0.000
f_j^3	0.000	0.000	0.000	0.000
f_j^4	0.000	0.000	0.000	0.000

4) Step 4: From Table IV, the criteria weights were calculated using Eq. (6). The results obtained are presented in Table VII.

TABLE VII. WEIGHT OF EACH CRITERION

Criterion	λ_1	λ_2	λ_3	λ_4
C_1	0.11	0.15	0.24	0.26
C_2	0.68	0.56	0.27	0.18
C_3	0.10	0.14	0.24	0.28
C_4	0.10	0.14	0.24	0.28

5) Step 5: Clustering coefficients were calculated for each study object and grey class using Eq. (7). As an example, the results obtained for O_1 study object are presented in Table VIII.

TABLE VIII. CLUSTERING COEFFICIENTS FOR O_1

Function	C_1	C_2	C_3	C_4	Coeff.
f_j^1	0.362	0.914	1.000	1.000	0.861
f_j^2	0.638	0.086	0.000	0.000	0.144
f_j^3	0.000	0.000	0.000	0.000	0.000
f_j^4	0.000	0.000	0.000	0.000	0.000

6) Step 6: Each study object was assigned to the class with the highest clustering coefficient, using Eq. (8). The results of maximum value of clustering coefficient for each study object are presented in Table IX.

TABLE IX. MAXIMUM VALUES OF THE CLUSTERING COEFFICIENT

Semester	λ_1	λ_2	λ_3	λ_4	Max.
2011-I	0.861	0.144	0.000	0.000	0.861
2012-I	0.853	0.136	0.058	0.000	0.853
2013-I	0.731	0.314	0.026	0.000	0.731
2014-I	0.816	0.167	0.083	0.000	0.816
2015-I	0.831	0.100	0.144	0.000	0.831
2016-I	0.833	0.189	0.000	0.000	0.833
2018-I	0.864	0.152	0.000	0.000	0.864
2019-I	0.857	0.155	0.000	0.000	0.857

IV. RESULTS AND DISCUSSION

The results and discussion, according to specific objectives in this work, are presented below:

A. Significance of the Grey Clustering Algorithm

This work contributes meaningful insights, both in theory and in practice, for classifying air quality in uncertain conditions. By applying the grey clustering algorithm (CTWF), which is based on the grey systems theory, it offers a flexible and adaptable way to classify air quality, especially in places where data are limited, a situation often seen in Latin American cities [24].

In addition, the grey clustering algorithm (CTWF) brings several practical advantages that make it highly applicable for managing air quality, as shown below:

- It delivers consistent and trustworthy results, even when working with small datasets or limited information, which is particularly useful for cities lacking extensive monitoring infrastructure [16].
- Unlike traditional air quality indices that typically reflect only the worst pollutant, the grey clustering algorithm (CTWF) provides a more complete picture by incorporating all measured pollutants into the evaluation [11].
- The use of triangular membership functions produces results that are easier to interpret for both technical experts and policymakers [14].

However, the grey clustering algorithm (CTWF) presents some limitations for considering, as shown below:

- It is focused on classification rather than prediction. Therefore, it needs to be complemented to design for forecasting or proactive environmental planning.
- It relies on predefined categories and thresholds, which can introduce some level of subjectivity into the analysis.
- While this algorithm used a specific method for weighting the importance of criteria, using different weighting schemes could lead to different classification outcomes [29].

Despite these limitations, this algorithm stands out as a robust and adaptable tool for analyzing how air quality changes over space and time. Its simplicity and resilience in contexts of uncertainty or limited information make it a valuable resource for environmental planning in fast-growing urban areas.

B. Performance of Algorithm in the Case Study

The grey clustering algorithm (CTWF), was implemented to classify air quality levels in Lima, Peru, over an eight-semester period from 2011 to 2019. The analysis was based on the concentration levels of four primary pollutants: PM_{10}, SO_2, NO_2, and CO. This classification relied on the calculation of grey clustering coefficients for each semester, allowing for an integrated assessment aligned with Peru's national Environmental Quality Standards (EQS).

As shown in Table IX, all semesters analyzed were categorized as "Good", with λ_1 clustering coefficients exceeding the 0.70 threshold. The highest air quality was recorded during the first semester of 2018, with a λ_1 value of 0.864. In contrast, the lowest value within the "Good" range was observed in the first semester of 2013, registering a λ_1 of 0.731. These findings suggest moderate seasonal variation, but an overall trend of compliance with national air quality standards.

Fig. 3 illustrates the temporal evolution of clustering coefficients. A gradual improvement in air quality is evident after 2016, likely reflecting the impact of stricter emissions regulations implemented in Lima's transportation sector. Conversely, the lowest λ_1 scores—ranging from 0.73 to 0.76—were observed during the first three semesters (2011-I to 2013-I), possibly due to the expansion of the vehicle fleet and increased industrial activity during that period [30].

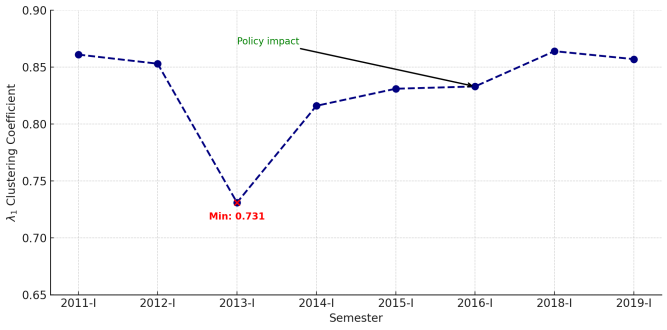


Fig. 3. Temporal evolution of air quality from 2011 to 2019.

To explore the behavior of individual pollutants, Fig. 4 presents a disaggregated analysis derived from the grey clustering results. Among all measured parameters, PM_{10} (C1) consistently recorded the lowest clustering values throughout the study period. In particular, during the first semester of 2013, NO_2 levels dropped below a clustering coefficient of 0.3, indicating poor air quality even though the overall classification remained within the "Good" category. These findings reinforce the characterization of PM_{10} as Lima's most persistent pollutant, primarily due to road dust re-suspension and emissions from diesel-powered vehicles [31].

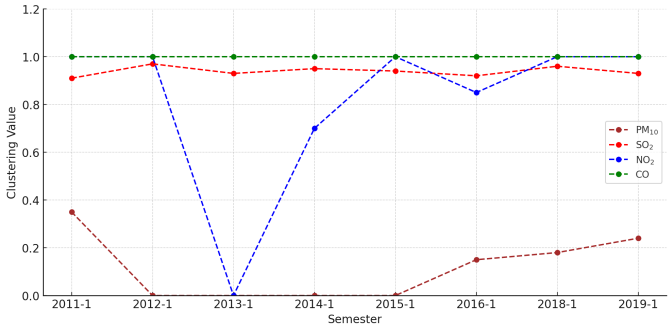


Fig. 4. Disaggregated results trends by pollutant from 2011-1 to 2019-1.

In contrast, CO (C_4) consistently recorded the highest grey clustering values—often above 0.9—indicating minimal health risk. Meanwhile, SO_2 (C_2) and NO_2 (C_3) showed moderate but stable trends. Their lower values during 2012 to 2013 may be linked to increased traffic emissions and reduced atmospheric dispersion capacity during dry seasons [6].

These temporal and pollutant-specific trends confirm the grey clustering algorithm (CTWF) strength in delivering comprehensive multi-criteria air quality assessments, even under conditions of uncertainty and limited information.

V. Conclusion

First, this work demonstrates that the grey clustering algorithm (CTWF) offers a practical and adaptable solution for classifying urban air quality under uncertain and limited information. Its capacity to integrate multiple pollutants highlights its suitability for real-world urban environments with restricted monitoring infrastructure. By enabling structured and interpretable assessments, the algorithm provides a reliable alternative to traditional models, contributing to more informed and inclusive environmental management strategies.

As a second key finding, the grey clustering algorithm not only enabled classification but also uncovered distinct pollutant-specific trends that would be masked by aggregated indices. While all periods met national "Good" air quality standards, disaggregated analysis revealed PM_{10} as the most recurrently critical pollutant, particularly in high-traffic and industrial areas. CO maintained consistently low risk levels, while SO_2 and NO_2 showed moderate but stable variability. Notably, a gradual improvement in air quality after 2016 suggests early effects of emissions control measures, although inconsistent enforcement likely limits their full potential. These insights underscore the value of multi-indicator, temporally sensitive approaches in supporting more nuanced and targeted environmental policy.

Finally, future research could expand the grey clustering algorithm (CTWF) by integrating meteorological variables

such as wind speed, temperature, and humidity to better contextualize seasonal pollution trends. The use of predictive grey models like GM(1,1) may strengthen early warning capabilities, while incorporating pollutants such as PM2.5 and ozone would improve health relevance. Combining CTWF with satellite-derived data could further enhance its spatial resolution and support its application in regional planning. As urban centers continue to grow under constrained conditions, grey modeling offers a promising pathway for more resilient and inclusive environmental governance.

REFERENCES

[1] World Health Organization, *Ambient (outdoor) air pollution*. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/ambient-(outdoor)-air-quality-and-health. Accessed: May 21, 2025.

[2] Y. Wang *et al.*, "The Effect of Long-Term Particulate Matter Exposure on Respiratory Mortality: Cohort Study in China," *JMIR Public Health Surveill*, vol. 10, p. e56059, Sep. 2024, doi: 10.2196/56059.

[3] K. Newell, C. Kartsonaki, K. B. H. Lam, and O. P. Kurmi, "Cardiorespiratory health effects of particulate ambient air pollution exposure in low-income and middle-income countries: a systematic review and meta-analysis," *Lancet Planet Health*, vol. 1, pp. e360–e367, 2017, doi: 10.1016/S2542-5196(17)30166-3.

[4] A. Manukova, K. Krastev, and I. Stoev, "Concentration for monitoring of air pollutants of domestic and industrial nature," in *Proc. 8th Int. Conf. Energy Efficiency and Agricultural Engineering (EE&AE)*, 2022, pp. 1–4, doi: 10.1109/EEAE53789.2022.9831404.

[5] I. C. Fernández, T. Koplow-Villavicencio, and C. Montoya-Tangarife, "Urban environmental inequalities in Latin America: A scoping review," *World Development Sustainability*, vol. 2, Jun. 2023, doi: 10.1016/j.wds.2023.100055.

[6] Y. Romero, N. Chicchon, F. Duarte, J. Noel, C. Ratti, and M. Nyhan, "Quantifying and spatial disaggregation of air pollution emissions from ground transportation in a developing country context: Case study for the Lima Metropolitan Area in Peru," *Science of the Total Environment*, vol. 698, Jan. 2020, doi: 10.1016/j.scitotenv.2019.134313.

[7] N. S. La Colla, S. E. Botté, and J. E. Marcovecchio, "Atmospheric particulate pollution in South American megacities," *Environmental Reviews*, vol. 29, pp. 415–429, 2021, doi: 10.1139/er-2020-0105.

[8] J. E. Johnston *et al.*, "Air quality and wheeze symptoms in a rural children's cohort near a drying saline lake," *Environmental Research*, vol. 263, Dec. 2024, doi: 10.1016/j.envres.2024.120070.

[9] A. A. Romero-Baylón, J. C. Mayorga-Rojas, J. C. Del Valle, W. J. Diaz-Cartagena, J. H. Ccatamayo-Barrios, and L. M. Soto-Juscamayta, "Spatial Localization of Air Pollutants in Lima: Air Quality Monitoring in the Troposphere," *International Journal of Environmental Impacts*, vol. 7, pp. 485–493, Sep. 2024, doi: 10.18280/ijei.070310.

[10] M. Hu and W. Liu, "Grey system theory in sustainable development research—a literature review (2011–2021)," *Grey Systems: Theory and Application*, vol. 12, no. 4, pp. 523–543, Nov. 2022, doi: 10.1108/GS-05-2022-0049.

[11] D. Julong and Y. Lin, "Introduction to grey systems theory," *Understanding Complex Systems*, vol. 68, pp. 1–24, 1988, doi: 10.1007/978-3-642-16158-2_1.

[12] N. Gouveia *et al.*, "Short-term associations between fine particulate air pollution and cardiovascular and respiratory mortality in 337 cities in Latin America," *Science of the Total Environment*, vol. 920, Apr. 2024, doi: 10.1016/j.scitotenv.2024.171073.

[13] N. Gouveia, A. D. Slovic, C. M. Kanai, and L. Soriano, "Air Pollution and Environmental Justice in Latin America: Where Are We and How Can We Move Forward?," Springer Science and Business Media Deutschland GmbH, Jun. 2022, doi: 10.1007/s40572-022-00341-z.

[14] A. Delgado, A. Aguirre, E. Palomino, and G. Salazar, "Applying triangular whitenization weight functions to assess water quality of main affluents of Rimac river," in *Proceedings of the 2017 Electronic Congress, E-CON UNI 2017*, 2018. doi: 10.1109/ECON.2017.8247308.

[15] A. Delgado and I. Romero, "Environmental conflict analysis on a hydrocarbon exploration project using the Shannon entropy," in *Proceedings of the 2017 Electronic Congress, E-CON UNI 2017*, 2018. doi: 10.1109/ECON.2017.8247309.

[16] A. Domenteanu, G. A. Crișan, C. Frăsineanu, and C. Delcea, "Exploring Grey Systems in Uncertain Environments: A Bibliometric Analysis of Global Contributions and Research Themes," *Sustainability (Switzerland)*, vol. 17, Mar. 2025, doi: 10.3390/su17062764.

[17] M. Gerus-Gościewska and D. Gościewski, "Grey systems theory as an effective method for analyzing scarce, incomplete and uncertain data on the example of a survey of public perceptions of safety in urban spaces," *Land*, vol. 10, no. 1, pp. 1–13, Jan. 2021, doi: 10.3390/land10010073.

[18] L. Zuo, S. Wang, Y. Sun, S. Cui, J. Fang, X. Ai, B. Li, C. Hao, and J. Wen, "Robustness assessment of wind power generation considering rigorous security constraints for power system: A hybrid RLO-IGDT approach," *CSEE J. Power Energy Syst.*, vol. 10, no. 2, pp. 518–529, 2024, doi: 10.17775/CSEEJPES.2023.05980.

[19] B. A. Alex Tume-Bruce, A. Delgado, and E. L. Huamaní, "Implementation of a Web System for the Improvement in Sales and in the Application of Digital Marketing in the Company Selcom," *Int. J. Recent Innov. Trends Comput. Commun.*, vol. 10, no. 5, pp. 48–59, May 2022, doi: 10.17762/IJRITCC.V10I5.5553.

[20] H. Lin, T. Pan, and S. Chen, "Comprehensive evaluation of urban air quality using the relative entropy theory and improved TOPSIS method," *Air Qual. Atmos. Health*, vol. 14, pp. 251–258, Feb. 2021, doi: 10.1007/s11869-020-00930-7.

[21] Q. Li, M. Ji, Y. Sun, X. Zhai, and Y. Zheng, "Analysis and evaluation of air quality in Shandong province based on AHP," in *IOP Conf. Ser.: Earth Environ. Sci.*, IOP Publishing Ltd, vol. 651, no. 4, Feb. 2021, Art. no. 042033, doi: 10.1088/1755-1315/651/4/042033.

[22] S. A. H. Morales, L. Andrade-Arenas, A. Delgado, and E. L. Huamaní, "Augmented Reality: Prototype for the Teaching-Learning Process in Peru," *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, pp. 806–815, 2022, doi: 10.14569/IJACSA.2022.0130194.

[23] A. Delgado, "Citizen criminality assessment in Lima city using the grey clustering method," in *Proc. 2017 IEEE XXIV Int. Conf. Electron., Electr. Eng. Comput. (INTERCON)*, Lima, Peru, 2017, pp. 1–4, doi: 10.1109/INTERCON.2017.8079662.

[24] J. A. Espinoza-Guillen, M. B. Alderete-Malpartida, F. D. Roncal-Romero, and J. C. Vilcanqui-Sarmiento, "Identification of particulate matter (PM10 and PM2.5) sources using bivariate polar plots and k-means clustering in a South American megacity: Metropolitan Area of Lima-Callao, Peru," *Environ. Monit. Assess.*, vol. 197, Mar. 2025, doi: 10.1007/s10661-025-13696-1.

[25] L. D. L. C. Rojas *et al.*, "Temporal Trend of PM10 and the Associated Risk to Human Health in the Lima Metropolitan Area," *Int. J. Sustain. Dev. Plan.*, vol. 20, pp. 1327–1335, Mar. 2025, doi: 10.18280/ijsdp.200338.

[26] J. J. Soria, A. O. Cardenas, and L. S. Peña, "Linear Regression with PM2.5 and PM10 Concentration for Air Quality in East Lima, Peru," in *Lecture Notes in Networks and Systems*, Springer Science and Business Media Deutschland GmbH, 2024, pp. 519–533. doi: 10.1007/978-3-031-70518-2_45.

[27] Organismo de Evaluación y Fiscalización Ambiental (OEFA), "Informe de Monitoreo," Lima, Peru, May 2024.

[28] MINAM, "Estándares de Calidad Ambiental (ECA) para Aire y establecen Disposiciones Complementarias," *El Peruano*, pp. 6–9, 2017. [Online]. Available: https://www.minam.gob.pe/wp-content/uploads/2017/04/Proyecto-de-DS-ECA-AIRE.pdf

[29] X. X. Xu, H. L. Gong, and X. Q. Ding, "Investigation and comparison of inertia weight control schemes in particle swarm optimization," in *Proc. 15th Int. Conf. Adv. Comput. Intell. (ICACI)*, 2023, pp. 1–8, doi: 10.1109/ICACI58115.2023.10146183.

[30] N. S. La Colla, S. E. Botté, and J. E. Marcovecchio, "Atmospheric particulate pollution in South American megacities," *Environ. Rev.*, vol. 29, pp. 415–429, 2021, doi: 10.1139/er-2020-0105.

[31] M. Bravo-Orellana, C. A. Bravo-Orellana, Á. R. Velázquez Fernández, and J. R. Pérez-Núñez, "Influence between chemical air pollution and cognitive abilities of school-age children in Metropolitan Lima," *Bordón. Revista de Pedagogía*, vol. 76, pp. 29–54, Dec. 2024, doi: 10.13042/Bordon.2024.102421.

Simulation-Driven Improvement of King Khalid University Non-Monthly Entitlement Workflows in AnyLogic

Elaf Ali Alsisi*, Osman A.Nasr, Badriah Mousa Alqahtani, Rodoon Shawan Alnajei
Business Informatics Department, King Khalid University, Abha, Saudi Arabia

***Abstract*—Aiming to offer useful recommendations for enhancing process accuracy and operational performance, this study investigates the elements affecting the efficiency and timeliness of disbursing non-monthly financial entitlements to university staff. The study developed a process workflow model using AnyLogic simulation tools and structured interviews and then conducted two main tests to enhance effectiveness. While the second concentrated on dynamic workload distribution to balance duties and maximize performance, the first conducted complete automation using an electronic platform that centralized all departmental tasks involved in financial disbursements. By lowering service times, minimizing manual errors, and thereby simplifying task allocation, the results show that combining automated workflows and real-time workload distribution significantly increased operational efficiency. Faster, more accurate, and fair processing of financial entitlements produced by this change highlights the need for technology-driven solutions in reaching lasting organizational excellence.**

Keywords*—Non-monthly financial entitlements; operational efficiency; Anylogic simulation; process optimization; university staff

I. Introduction

Higher education institutions' foundation is administrative procedures, which help to define their capacity to provide services, research, and high-quality education. From policy development and resource allocation to performance assessment and decision-making, these procedures cover a broad spectrum of activity. Good administrative systems guarantee that colleges may adjust to fast-changing educational environments, simplify processes, and promote organisational effectiveness. Strong administrative policies in higher education help to increase openness, responsibility, and resource optimization, thereby improving institutional performance, according to Johnson et al. (2021). Furthermore, these procedures help to match operational frameworks with strategic goals, thereby fostering a coherent environment for staff, teachers, and students [1].

Administrative process efficiency—especially those pertaining to cash disbursement—depends on several important elements. These cover data availability and accuracy, system degree of automation, and internal policy and procedure clarity. Also very important are outside factors such as economic conditions and legal constraints. Inefficiencies in financial disbursement systems can cause delays, mistakes, and employee discontent that finally influence organizational performance, as Smith and Brown (2020) point out. Thus, increasing efficiency and guaranteeing the timely and accurate distribution of financial advantages depend on an awareness of and resolution of these elements [2].

Non-monthly entitlements are connected to non-periodic financial remuneration paid either monthly or otherwise. These rights include incentives given to staff members on sporadic intervals and financial gains. The several types of non-monthly entitlements include unique incentives, spot awards, and annual bonuses. Understanding their influence on efficiency and the speed of payment depends on first looking at the factors influencing performance and staff satisfaction in connection with these categories. The results of Green et al. (2022) show that prompt and open handling of non-monthly entitlements can greatly increase employee morale and involvement [3].

The sensitive process of distributing non-monthly entitlements to university staff members calls for great satisfaction to guarantee operational success. The disbursement process is influenced by internal elements such the effectiveness of administrative procedures and technological integration as well as by external elements including regulatory compliance and financial restraints. Improving the process depends on an awareness of and analysis of these elements, therefore optimising the advantages obtained by university staff members. This therefore promotes a good workplace, so improving organisational performance and helping to meet institutional objectives.

This study, therefore, aimed to investigate how modeling and simulation using AnyLogic contribute to the analysis of both internal factors, such as the efficiency of administrative procedures and technology integration, and external factors, including financial constraints and regulatory requirements, that affect the process of disbursing non-monthly financial benefits to university employees. It also sought to identify the root causes of delays and inefficiencies while evaluating the impact of proposed interventions—such as automation, improved workload distribution, and enhanced intercommunication—on disbursement speed, accuracy, and employee satisfaction.

This research is structured as follows: Section I provides an overview of the topic. Section II presents the problem statement and objectives. While Section III reviews the relevant literature, the following part describes the methodology and data collection procedures used to conduct the analyses in Section IV. Section V synthesizes the key findings from both experiments and introduces the simulation model used to assess the underlying mechanism. The final section summarises the principal findings

*Corresponding Author.

of these experiments and includes a discussion of the implications of the findings for future research.

II. Problem Definition

A. Problem Statement

The university's disbursing non-monthly financial obligations presents many difficulties and delays. These issues undermine administrative performance and personnel satisfaction. Although these dues help to improve employee happiness, there is no thorough investigation to pinpoint the underlying reasons for these issues.

B. Objectives

This study intends to use AnyLogic simulation and modelling techniques to investigate the several elements affecting the efficiency and timeliness of disbursing non-monthly entitlements to university staff. Investigating the fundamental reasons and variables influencing the disbursement process, the study looks at how they affect work efficiency and production. Furthermore, investigated in the study are possible fixes for found inefficiencies like automation, workload distribution optimisation, and improved interdepartmental communication.

III. Literature Review

Maintaining excellent service standards, colleges have to take care of environmental issues and guarantee administrative effectiveness just like any other company. Among the several aspects of educational excellence, administrative process management has lately attracted major interest [4]. Most facets of human life have changed dramatically in recent decades thanks to the fast advancements in global markets and scientific knowledge and technological innovation [5], [6], [7]. Equipping students with the information and the skills needed to fulfil changing labour market needs depends critically on universities [8].

Delivery of high-quality university services, including teaching and learning, depends fundamentally on administrative processes [9]. Beyond instruction, administrative processes are crucial for controlling student services, offering direction, and preserving infrastructure quality, Manatos et al. underline. Using quality control strategies in these spheres guarantees a comprehensive academic background [10]. Good internal management improves daily operations and helps institutions to provide outstanding services in several areas including student assistance, infrastructure, assessment systems, and facilities management [11]. Moreover, a key indicator of administrative service efficacy is student satisfaction [12].

The administrative structures of higher education institutions significantly impact their overall performance and operational efficiency. Strong administrative processes are emphasised by Johnson et al. (2021) as a means of promoting openness, responsibility, and resource economy [13]. Perez et al.'s 2020 research emphasises how digital platforms and automation help to expedite administrative processes. These tools streamline decision-making, reduce errors, and optimize resource allocation [14]. Further influencing the effectiveness of financial systems inside public institutions are external regulatory constraints and economic situations [15], [16].

As non-monthly rights, bonuses and incentives are quite important for raising the morale and involvement of administrative staff members. But uneven distribution of these benefits has been proven to lower employee satisfaction and productivity, mostly because of inefficiencies in financial clearances among several departments [17]. Neglect of non-monthly rights could cause more general organisational problems like higher staff turnover and worse institutional efficiency [18].

Complex workflow optimisation has benefited from simulation and modelling tools such AnyLogic. Since 2003, AnyLogic—a flexible tool for modelling discrete, continuous, and mixed-agent behaviour has been used across manufacturing, supply chain management, logistics, retail, traffic systems, aerospace, and manufacturing [19]. Kim et al. (2020) show how financial process improvements suggested by simulation models might point up roadblocks [20]. Though modelling approaches are widely used in many fields, their application within higher education management—especially in financial systems—remains understudied. There is a clear gap in the literature since little studies specifically address non-monthly financial entitlements in universities. This emphasises great possibilities to use advanced AI-based technologies to boost administrative efficiency and enhance money distribution methods.

IV. Data Collection and Methodology

A. Data Collection

Structured interviews were the main means of data collecting for this research; they were the main tool used to compile qualitative data. Carefully crafted to ensure they gathered comprehensive and practical data relevant to the study goals, the interview questions [21] Particularly focused on investigating the procedures behind financial transactions, the flow of labour across departments, and the difficulties these changes bring about was The methodical approach of the interviews allowed participants to offer in-depth observations based on their operational expertise while nevertheless allowing consistency in data collecting.

Selected staff members from two important departments—the Payroll Department and the Human Resource Operations Department—conducted these interviews. These interviews were designed to thoroughly understand the current procedures, identify operational difficulties, and compile information relevant to the scope of this research.

At King Khalid University, the data collection process took place during regular business hours to coincide with the participants' natural flow and minimize interruptions to their schedules. A purposive sampling method was used to target people directly involved in the under-investigated processes. Ten transactions in all were chosen as samples for analysis to provide a fair picture of university financial transaction handling procedures.

Every interview took a one-on-one form and lasted between thirty and forty-five minutes. Before the interview, participants were briefed on the goal of the study and guaranteed informed consent and were encouraged to provide honest answers.

B. System Design

Modelling and simulation are the main tools used in this work; the simulation model was created with the AnyLogic application. The study starts with offering a conceptual model of the university's financial entitlement dispensing system in order to set the scene. Before switching to simulation-based modelling, this conceptual model helps to clarify every stage of the system and offers complete knowledge of its process [22].

The two main operational entities in charge of starting and finishing money distributions comprised the system:

- Human resources operations department: represented by one server.
- Payroll department: represented by one server.

For every transaction, transaction record data was used to construct pertinent random variables, therefore guaranteeing correct simulation results. Among these were:

- Average waiting time: The duration a request remains in the queue before being processed.
- Average service time: The time required by each department to complete a single request.
- Average time to issue a payment order: The total duration from the initiation of the transaction to the issuance of the payment order [22].

C. System Configuration

Two studies evaluating the financial disbursement process made use of AnyLogic simulation tools. Under specified operational conditions, these tests gave important new perspectives on the efficiency and performance of the system, enabling a thorough investigation of waiting times, processing times, and general transaction completion times. Conducted throughout regular business hours, five weekdays from 8:00 AM to 2:30 PM, the simulations were based on the presumption of a limitless queue capacity.

1) First experiment: The first trial was meant to evaluate how well automated processes implemented in several departments engaged in the disbursement process worked. This experiment sought to find whether automation might cut processing times, simplify processes, and lower mistake rates.

2) Second experiment: The second experiment concentrated on bettering staff member workload distribution at every level of the process. This experiment aimed to improve operational efficiency and lower delays resulting from unequal workload distribution or bottlenecks by best allocating jobs.

3) Simulation assumptions: Four transactions arrive at regular intervals of every half hour, hence defining the transaction arrival rate. The simulation is based on these assumptions stated in Table I. Two employees are assigned to the Operations Department and two employees to the Payroll Department, therefore allocating the manpower. One hour after arrival, every department starts handling a transaction to guarantee a neat flow. Assigning transactions to the staff member not handling other transactions helps to distribute work fairly and reduces idle time [23].

4) Simulation framework: The simulation model was built to depict a queuing mechanism for the disbursement of non-monthly financial entitlements [23]. To find inefficiencies and bottlenecks, the model combines important elements of the system and models the flow of transactions between departments.

5) Queuing system classification: Kendall's notation for classifying queuing systems helps one to define this system as having:

- Multiple queues with independent, parallel paths for transaction processing.
- A queue discipline that adheres to the First-In, First-Out (FIFO) rule, ensuring fairness in processing requests.

The simulation model well reflects the dynamics of the system, thereby allowing the research to assess the performance criteria and, where needed, process optimisation [22].

6) Arrival process: The arrival of transactions is modelled as a random process, with the interarrival time (A-i) defined as the time difference between the arrivals of the (i-1) th and the i-th transactions [24].

$$(A\text{-}i = T\text{-}i \; - T\text{-}\{i\text{-}1\}) \tag{1}$$

The mean interarrival time is denoted as E(A), and the arrival rate (λ) is calculated as:

$$\lambda = 1 / E(A) \tag{2}$$

For this study, we assume transactions arrive every 3 hours, so:

E(A) = 3 hours, λ = 1/3 = 0.333 transactions per hour.

TABLE I. SIMULATION ASSUMPTIONS FOR WORKFLOW OPTIMIZATION IN NON-MONTHLY ENTITLEMENT PROCESSING

Tra No.	Transaction Type	OD	PD	AT*	SS - OD	SE - OD	WT - OD	SS - PD	SE - PD	WT - PD	Total Service Time
1	Extra Allowance Request	E1	E1	9:00 AM	10:00 AM	01:00 PM	3 hrs	2:00 PM	9:00 AM	1 hrs	24 hrs
2	Financial Dues Request	E2	E2	9:30 AM	10:30 AM	01:30 PM	3 hrs	2:30 PM	9:30 AM	1 hrs	24 hrs
3	Excellence Allowance	E1	E1	10:00 AM	11:00 PM	2:00 PM	3 hrs	9:00 AM	1:00 PM	19 hrs	27 hrs
4	Conference Attendance	E2	E2	10:30 AM	11: 30PM	8:30 AM	9:30 hrs	9:30 AM	1:30 PM	1 hrs	27 hrs

*(OD) Operations department, (PD) Payroll department, (AT) Arrival Time, (SS) Service Start Time, (SE) Service End Time, (WT) Waiting Time.

V. RESULTS

The study concentrated on the university's disbursing non-monthly financial obligations, including conference-related financial reimbursements and excellence allowances. This system runs by means of a cooperative process combining several departments. Working closely with the Payroll Department and the Financial Department, the Human Resources Operations Department is key in starting and processing transactions. These departments taken together guarantee the seamless flow of the process and preserve the accuracy and integrity of financial disbursements.

As Fig. 1 shows, the payout process consists of multiple consecutive processes carefully planned to guarantee accuracy and conformity with institutional standards. It starts with the transaction submission, therefore starting the process. The issuing of a disbursement decision, formalising the approval for payment, the creation of a disbursement statement detailing the payment specifics, and lastly, the actual distribution of funds to the intended recipient follow.

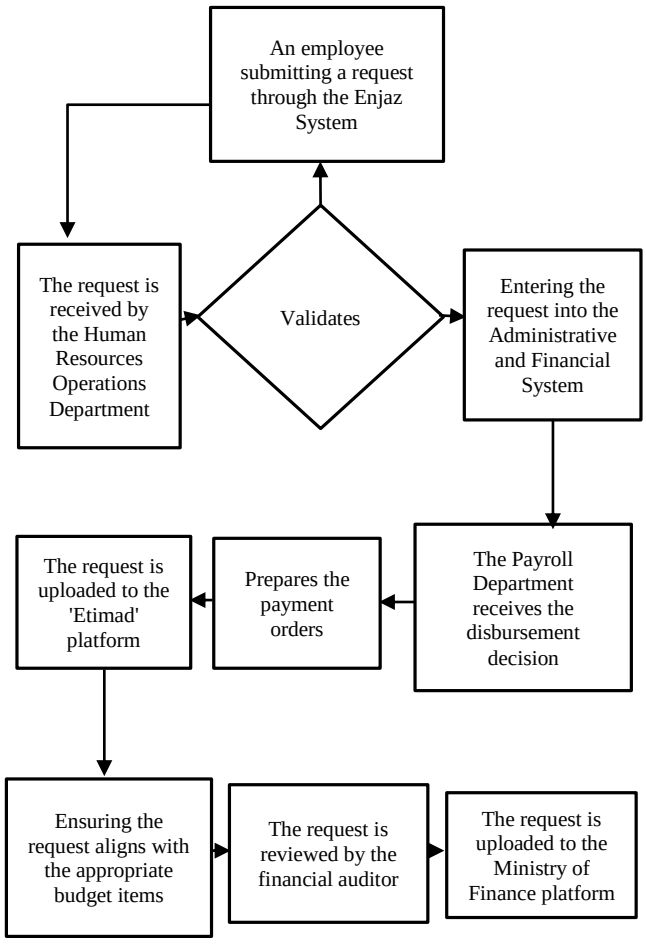


Fig. 1. Workflow figure model for non-monthly entitlement processing.

Every one of these phases entails different administrative processes and is influenced by different elements, such policy adherence, documentation correctness, and interdepartmental collaboration. Reliability and efficiency of the system depend on these components, which are essential.

The results of the organised interviews exposed a clear awareness of how the disbursement of entitlements operates the workflow. The system's workflow concept is meant to simplify processes, lower mistakes, and lessen inefficiencies. The participants underlined the need of interdepartmental cooperation and following accepted policies in reaching a flawless and efficient payout procedure.

The information gathered from the interviews contained specifics on ten transactions from which the departments engaged in the disbursement process may examine waiting times and service times. The performance of the operations department and the payroll department showed notable variations according to the study of service time and waiting time rates.

With an average service time (μ) of 115 hours and a service rate (λ) of over 0.0087 services per hour, the operations department has

$$\Pi\ (T \leqslant 50) = 35.26\% \quad (3)$$

$$\Pi\ (T \leqslant 100) \approx 58.09\% \quad (4)$$

The likelihood of executing a transaction within 50 hours is

P ($T \leqslant 50$) whereas inside 100 hours is P ($T \leqslant 100$).

With an average service time (μ) of 73 hours and a service rate (λ) of over 0.0137 services per hour, the payroll Department exhibits a notably improved performance. P($T \leqslant 50$) = 49.59% and P ($T \leqslant 100$) $\approx$ 74.59%, respectively, respectively, the chances of handling a transaction inside 50 and 100 hours.

With a waiting time rate (λ) of over 0.0278 per hour, the Operations Department averages a waiting time (μ) of 36 hours.

With a waiting time rate (λ) of 0.3333 per hour and an average waiting time (μ) of 3 hours, the payroll department shows a far shorter waiting time.

These results underline the payroll department's higher processing efficiency than the operations department, thereby stressing the requirement of focused interventions to maximise the workflow in the latter to avoid congestion and raise general efficiency.

VI. DISCUSSION

Differences in service indicate an organisational flow imbalance: slower processing times in the operations department can create transaction backlogs, causing further delays in other departments. These inefficiencies pose the potential of impairing the disbursement system's overall performance. Optimising service rates across several departments is essential to meet these difficulties. Workflow could be balanced by means of reallocating resources, automating repetitive processes, or offering people in the operations section focused training. Improving the effectiveness of the operations department will help the company to match its procedures more precisely with institutional goals, smooth operations, and increase productivity.



Fig. 2. Simulation model of the financial disbursement workflow.

Incorporating thorough methods and characteristics to precisely mirror real-world settings, the simulation model was constructed to analyse and solve the found inefficiencies as shown in Fig. 2. The first experiment concentrated especially on assessing the success of using automated processes among several divisions.

The suggested fix was substituting a centralised computerised platform for the hand-operated disbursing of financial obligations. Using a single interface, this system let administrative and academic staff members send non-monthly financial dues requests straight forwardly. To provide a more simplified and open procedure, requests were immediately passed to the necessary management for clearance. This system had a filter mechanism to confirm the completeness of turned in forms. While incomplete requests were automatically denied, therefore lowering delays resulting from incomplete entries, requests satisfying all criteria were sent straight to the Human Resources Operations Department for additional processing.

The usual transaction processing time before the automated system was 188 hours. After the new workflow was adopted and system analysis was conducted with the simulation model (Fig. 3), the average service time dropped to 59 hours per transaction rather dramatically. This shows an amazing drop of about 69%, which emphasises how well the system maximises operations.



Fig. 3. Impact of workflow automation on average service time.

The significant drop-in service time shows the good influence of process optimisation and automation. Requests were centralised and automated validation introduced helped to streamline the process, therefore minimising human mistakes and processing delays. The simplified procedure also helped to better allocate resources, enabling departments to manage transactions more precisely.

The second experiment concentrated on improving operational efficiency by means of better distribution of staff duty at every level. Based on their real-time workload, the suggested solution presented a dynamic assignment of requests to accessible staff members. This invention reduced delays brought on by waiting for staff members to handle jobs, therefore removing the need for manual job allocation.

Once a request is authorised at the first level, the automated system guarantees that it is immediately assigned to the next most qualified employee in the pertinent department. This procedure reduces idle time, ensures a fair distribution of chores, and enhances the overall processing efficiency of non-monthly financial dues. Distribution of work dynamically helps the system to minimise congestion caused by overloading stages or staff. The findings of the queue simulation show notable increase in operational efficiency. Every transaction's computed average service time was 25.50 hours. This suggests a simplified procedure that helps to handle individual transactions faster than past methods.

The efficacy of the automated workload distribution system in besting the task allocation among staff is shown in the shortened service times. Reducing delays at different phases guarantees that every transaction is handled effectively without generating congestion. This balanced workload enhances the flow of operations among departments, therefore enabling better use of resources and a more responsive system. Along with accelerating transaction processing, this development improves the general dependability and sustainability of the distribution mechanism. The results underline the need of dynamic task distribution systems in reaching operational excellence and guaranteeing timely service delivery.

From the first reception of requests to their flawless transfer across the operations and payroll departments, the use of an electronic platform for conducting financial transactions has fundamentally changed the payout process. Establishing itself as a pillar of operational efficiency, this creative approach has produced observable gains in various important spheres.

The shortened waiting times are one of the most obvious results since the platform replaced slower manual operations with a simplified, technologically driven workflow. This development not only sped up transaction processing but also cut pointless delays. Moreover, the flawless handover of transactions between departments and automatic work allocation helped to improve efficiency by guaranteeing better operations and so lowering of obstacles. Reducing mistakes, handling the difficulties with hand data entry, and improving interdepartmental communication all depend on the platform as well. More accurate and dependable data handling produced by these developments enhanced the general integrity of the system.

Beyond these operational improvements, the platform has raised staff satisfaction. Faster processing speeds and more openness helped build system trust and hence increase staff confidence in its dependability. Furthermore, the platform's capacity to provide performance monitoring by means of transaction completion rates and processing times has helped to precisely track development and highlight areas needing continuous enhancement.

VII. Conclusion

By greatly boosting speed, accuracy, and efficiency and thereby creating a more fulfilling and transparent experience for staff, the electronic platform has transformed the payout process. Combining technology and automation has shown its transforming power in solving long-standing inefficiencies and improving operational processes. These developments not only improve current operations but also provide a strong basis for reaching sustained operational excellence and therefore

supporting the long-term expansion goals of the company. The findings underscore the pivotal role of technology-driven solutions in improving organisational performance and provide a compelling case for continued investment in automation and process innovation to maintain and extend these gains.

Building on the ~69% reduction in processing time and the observed Payroll- Operations service-rate gap, future work should scale the centralised platform while using simulation–optimisation to right-size Operations capacity and routing. To sustain and generalise the gains, future studies should validate the redesigned workflow across other non-monthly entitlements and semesters, and deploy a digital twin that forecasts backlogs and recommends weekly staffing plans.

ACKNOWLEDGMENT

The authors extend their appreciation to the Deanship of Research and Graduate Studies at King Khalid University for funding this work through Genersal Research Project under grant number GRP/112/46.

REFERENCES

[1] T. Johnson & Lee, P., "Administrative excellence in higher education: Strategies for improvement," *Journal of Higher Education Administration*, vol. 34, no. 2, pp. 123–138, 2021.

[2] R. Smith & Brown, S., "Financial disbursement systems in universities: Challenges and opportunities.," International Journal of Educational Management, vol. 28, no. 3, pp. 567–582, 2020.

[3] A. Green et al., "Employee incentives and organizational performance: A case study of higher education institutions," Human Resource Development Review, vol. 15, no. 4, pp. 456–478, 2022.

[4] D. Alvarez-Sández, K. Velázquez-Victorica, A. Mungaray-Moctezuma, and A. López-Guerrero, "Administrative Processes Efficiency Measurement in Higher Education Institutions: A Scoping Review," *Education Sciences 2023, Vol. 13, Page 855*, vol. 13, no. 9, p. 855, Aug. 2023, doi: 10.3390/EDUCSCI13090855.

[5] S. Nadeau, "Lean, Six Sigma and Lean Six Sigma in Higher Education: A Review of Experiences around the World," American Journal of Industrial and Business Management, vol. 07, no. 05, pp. 591–603, 2017, doi: 10.4236/ajibm.2017.75044.

[6] S. M. Christopher, "Participatory monitoring and evaluation for quality programs in higher education: What is the way for Uganda?," International Journal of Educational Administration and Policy Studies, vol. 12, no. 1, pp. 52–59, 2020, doi: 10.5897/ijeaps2020.0637.

[7] J. Texeira-Quiros, M. do R. Justino, M. G. Antunes, P. R. Mucharreira, and A. de T. Nunes, "Effects of Innovation, Total Quality Management, and Internationalization on Organizational Performance of Higher Education Institutions," Front Psychol, vol. 13, p. 869638, Apr. 2022, doi: 10.3389/fpsyg.2022.869638.

[8] D. E. Salinas-Navarro, C. Mejia-Argueta, L. Montesinos, and E. Z. Rodriguez-Calvo, "Experiential Learning for Sustainability in Supply Chain Management Education," Sustainability (Switzerland), vol. 14, no. 20, p. 13133, Oct. 2022, doi: 10.3390/su142013133.

[9] R. O. Vrielink, V. Nijhuis-Boer, C. van Horne, E. Hans, and J. van Hillegersberg, "A framework for planning and control of the education organization," in Smart Innovation, Systems and Technologies, Springer, Cham, 2019, pp. 234–245. doi: 10.1007/978-3-319-92363-5_22.

[10] M. J. Manatos, C. S. Sarrico, and M. J. Rosa, "An integrative approach to quality management in higher education?," TQM Journal, vol. 29, no. 2, pp. 342–356, 2017, doi: 10.1108/TQM-01-2016-0009.

[11] R. N. Bin Rais, M. Rashid, M. Zakria, S. Hussain, J. Qadir, and M. A. Imran, "Employing industrial quality management systems for quality assurance in outcome-based engineering education: A review," Jan. 26, 2021, Multidisciplinary Digital Publishing Institute. doi: 10.3390/educsci11020045.

[12] A. Valencia-Arias et al., "Model Proposal for Service Quality Assessment of Higher Education: Evidence from a Developing Country," Educ Sci (Basel), vol. 13, no. 1, p. 83, Jan. 2023, doi: 10.3390/educsci13010083.

[13] A. Johnson Smith, B., & Taylor, C., "Optimizing administrative efficiency in higher education institutions: A framework for transparency and accountability," Journal of Educational Management, vol. 45, no. 3, pp. 234–256, 2021.

[14] D. Perez Gonzalez, L., & Martinez, J., "The role of automation in higher education administration: A review of digital transformation strategies," Education Technology Review, vol. 32, no. 2, pp. 45–60, 2020.

[15] R. Smith & Brown, P., "Regulatory impacts on financial management in public universities: A comparative study.," Public Administration Quarterly, vol. 38, no. 2, pp. 67–89, 2020.

[16] R. Smith & Brown, P.Smith, R., & Brown, P., "Regulatory impacts on financial management in public universities: A comparative study," Public Administration Quarterly, vol. 38, no. 2, pp. 67–89, 2020.

[17] E. Green White, R., & Harris, M., "Employee satisfaction and financial incentives: Implications for organizational success," Human Resource Studies, vol. 56, no. 4, pp. 789–805, 2022.

[18] M. Mudhofar, "The Effect Of Performance Allowances On Motivation, Employees' Work Achievement and Organizational Performance in Government Offices," Jurnal Administrasi Publik : Public Administration Journal, vol. 11, no. 1, pp. 101–110, Jun. 2021, doi: 10.31289/jap.v11i1.4963.

[19] B. Zlatanovska, N. Stojkovikj, M. Kocaleva, A. Stojanova, L. K. Lazarova, and R. Gobubovski, "Modeling of some chaotic systems with anylogic software," TEM Journal, vol. 7, no. 2, pp. 465–470, May 2018, doi: 10.18421/TEM72-31.

[20] J. Kim Lee, S., & Park, K. (2020), "Simulation and modeling approaches for process optimization in financial systems," Operations Research Letters, vol. 46, no. 6, pp. 98–113, 2020.

[21] T. Melani and Y. Iriani, "Analysis and Modeling of Queuing System Simulation in Payment Process at Minimarket (Case Study of Minimarket X Yogyakarta)," Jurnal Indonesia Sosial Teknologi, vol. 5, no. 5, 2024, [Online]. Available: http://jist.publikasiindonesia.id/

[22] S. Dutkova, K. Achimsky, and P. Drozdziel, "Simulation of a queuing system of a post office in anylogic software," Communications - Scientific Letters of the University of Žilina, vol. 22, no. 2, pp. 15–22, Jan. 2020, doi: 10.26552/com.C.2020.2.15-22.

[23] P. Umarani and S. Shanmugasundaram, "A Study on M/M/C Queueing Model under Monte Carlo Simulation in a Hospital," 2016. [Online]. Available: http://www.ripublication.com

[24] M. A. Ghaleb, U. S. Suryahatmaja, and I. M. Alharkan, "Modeling and simulation of Queuing Systems using arena software: A case study," in IEOM 2015 - 5th International Conference on Industrial Engineering and Operations Management, Proceeding, Institute of Electrical and Electronics Engineers Inc., Apr. 2015. doi: 10.1109/IEOM.2015.7093945.

From Review to Refinement: An Expert-Informed Environmental Diagnostic Model for Stingless Bee Colony Monitoring

Yang Lejing[1], Rozi Nor Haizan Nor[2*], Yusmadi Yah Jusoh[3], Nur Ilyana Ismarau Tajuddin[4], Khairi Azhar Aziz[5]
Department of Software Engineering and Information System, Faculty of Computer Science and Information Technology
Universiti Putra Malaysia[1, 2, 3]
Tamhidi Centre, Universiti Sains Islam Malaysia[4]
IT Enterprise Architecture, IT Strategy, Bank Kerjasama Rakyat Malaysia[5]

***Abstract*—The resilience of stingless bee colonies has become increasingly challenged by erratic climate conditions and intensified environmental stressors. While previous studies have introduced diagnostic models for monitoring colony health, most remain constrained by a narrow reliance on either environmental or behavioral parameters alone. This study proposes a refined diagnostic model that builds on existing frameworks and is further shaped by expert insights from the field. The model integrates environmental inputs, specifically temperature and humidity, with behavioral activity detected via video analysis to deliver a multi-dimensional assessment of colony status. Through a structured review of the literature and interviews with apiculture experts, we identify critical gaps in conventional systems and translate those findings into a more responsive and field-deployable architecture. The result is an improved model capable of categorizing colony health with greater sensitivity and clarity, designed to support early intervention and long-term monitoring. The model is visualized through comparative schematic diagrams, showing the evolution from a basic environmental-only logic to a more holistic decision-making system.**

Keywords*—*Stingless bees; environmental monitoring; behavioral analysis; diagnostic model; expert-informed refinement

I. INTRODUCTION

Stingless bees are considered one of the most important pollinators in tropical and subtropical regions [1]. They can help in sustaining crop production and balancing ecological systems [2], especially in areas where traditional pollinators like honeybees are in decline. Unlike other bees, stingless bees are more adaptable and are kept in small-scale meliponiculture farms for honey, pollen, and propolis. The existence and health of these colonies impact not only biodiversity but also the economic welfare of rural populations who rely on these products.

Nonetheless, the existence of stingless bee colonies is increasingly threatened by sudden environmental changes, such as erratic temperature changes, long periods of high humidity, and habitat destruction [3]. These stressors often weaken colonies and, in more extreme cases, can lead to colony collapse disorder, in which bees suddenly and inexplicably abandon the hive.

The most common approach to assess the colonies' health remains manual inspection. These methods usually require opening the hive and examining stores of food, brood patterns, and bee activity; however, frequent inspections are invasive and can disturb the colony's microclimate and social thermoregulation, thereby increasing stress on the colony. Moreover, manual checks are typically episodic and depend on the observer's experience and judgment, making them subjective and prone to inconsistency; as a result, important early-warning signals may be missed or detected late. Recent reviews, therefore, highlight the growing role of non-invasive, sensor-based precision-beekeeping and AI methods as complementary tools to reduce disturbance and provide continuous, objective monitoring [4], [5], [6], [7].

To address these problems, technological solutions have been gradually integrated into beekeeping to enable more continuous and less invasive monitoring. However, many existing precision-beekeeping systems primarily rely on a narrow set of environmental parameters, most commonly in-hive and ambient temperature, humidity and sometimes hive weight, measured by simple sensors. While these indicators are valuable for assessing thermostatically performance and colony strength, they do not directly capture behavioral responses such as reduced movement, altered foraging flux, entrance traffic changes, or other anticipatory actions that can precede overt environmental shifts. Consequently, systems that omit behavioral dimensions are limited in their ability to detect early warning signals and critical precursors of colony stress; recent literature therefore recommends integrating video, audio, weight and advanced pattern-recognition methods to improve early diagnostics [8], [9], [10].

Existing stingless bee monitoring systems predominantly focus on either environmental sensing or activity observation in isolation. Environmental-only systems therefore provide useful information about nest microclimate or mass balance, but they are limited in detecting early behavioral responses, such as changes in movement, entrance traffic, or foraging flux, that often precede measurable environmental shifts. Conversely, behavior-only approaches (camera or acoustic monitoring) typically lack concurrent environmental context, which complicates interpretation of causal drivers. In addition, many operational threshold values and alarm rules have been

inherited or adapted from Apis-centric studies without systematic validation for Meliponini species, reducing predictive accuracy when applied to stingless-bee colonies in varied ecological settings. Finally, several deployed systems use monolithic, fixed architectures that are difficult to extend to additional sensor modalities or site-specific calibration, constraining adaptability across management regimes and habitats. Recent reviews and empirical studies, therefore, recommend multimodal, modular monitoring architectures and species-specific validation of thresholds to improve early-warning performance for stingless bee management [11], [12], [13], [14].

The motivation for this research arises from the need to bridge these gaps through an integrated, adaptable, and empirically validated diagnostic framework. This study proposes a model that combines temperature and humidity monitoring with automated behavioral analysis using computer vision, refined through expert consultation to ensure parameter thresholds align with local field realities. By embedding the system within a modular architecture, it supports future expansion to include additional environmental or behavioral indicators as required by practitioners.

This study aims to address the following research questions:

1) How can environmental monitoring and bee activity analysis be integrated into a unified diagnostic framework for stingless bee colony health?

2) What key environmental and behavioral parameters are critical to accurately classify colony health status?

3) How can expert validation be incorporated to refine the model and enhance its practical applicability in real-world beekeeping?

The remainder of this study is organized as follows: Section II reviews relevant literature on stingless bee monitoring systems and related technologies. Section III presents the proposed initial model, outlining its structure and components. Section IV describes the expert review process and the subsequent model refinements. Section V reports the results and analysis of expert feedback. Section VI discusses the findings, identifies limitations, and suggests directions for future research. Section VII concludes the study.

II. Literature Review

The deterioration in stingless bee health is frequently signaled by declining population levels, decreased foraging activity, and irregular behavioral patterns. These symptoms often result from the complex interplay between environmental disturbances and biological stress [1]. Fluctuating temperature and humidity, increasing pollutant exposure, and unpredictable climate variability have been identified as key contributors to destabilizing colony health. As an example, sudden temperature fluctuations can disrupt bees' navigation systems, causing lower foraging efficiency and disorientation on the way back to the hive [15]. Pesticides and heavy metals introduced to the food chain further weaken immune systems, diminishing the vitality of colonies [3].

Due to this ecological fragility, stingless bees are viewed as effective indicator of broader environmental changes. Among all environmental variables, temperature is by far the most important. It controls the metabolic rate, enzyme activity, and developmental processes within colonies. Departures from ideal temperature ranges either above or below will disturb normal physiological functions, impair social cohesion, and increase susceptibility to infections [16]. Excessive heat can cause high water loss, energy loss, and loss of internal hive control, while the cold slows metabolism and essential cleaning tasks vital to hygiene. At the same time, relative humidity affects the microclimate important for rearing brood and for the integrity of wax. These reasons explain why the parameters are considered essential indicators for assessing the impact of human activity on the environment and monitoring the health of the colony.

Monitoring of stingless bees has improved with the incorporation of technology. Real-time environmental monitoring, such as the Internet of Things (IoT), can utilize temperature, humidity, gas, and even sound sensors to check on health systems [17]. Though these sensor technologies facilitate continuous monitoring, early detection of risks, and automated data collection, their practical application is still hampered by significant obstacles like integration difficulties, drifting sensors, and unreliable power supplies.

Modeling environmental risks has received great focus because of its ability to predict. Many models combine several environmental factors like temperature, humidity, and air conditioning to create risk assessment models that can provide dynamic alerts and automation in decision-making [18]. Advanced computation methods, like statistical methods and machine learning, have developed these models. Even with the availability of data, there is still a challenge in creating usable models that are interpretable for the field in different ecological contexts. In response, recent systems have integrated web-based dashboards for real-time sensor data visualization using Flask and similar frameworks [19]. While these platforms improve usability, their performance depends on reliable connectivity, efficient data processing, and interface responsiveness, areas still under active development [20].

Growing interest in bee rehabilitation has driven parallel innovation in system architectures. Several projects have applied LoRa communication protocols to enable wide-area data collection in low-power field settings [21]. Predictive frameworks based on real-time sensor data have also been proposed to mitigate health risks [22]. Beyond colony-level monitoring, studies on plant resource availability underscore the broader ecological factors affecting bee behavior and resilience [23]. A system-based approach for understanding colony decline recommends modeling across behavioral, environmental, and management layers [24], aligning well with recent IoT-enabled diagnostic platforms [25].

Stingless bee research increasingly emphasizes smart hive solutions. Low-cost microcontrollers combined with DHT22 temperature sensors and weight modules have formed the basis for continuous environmental assessment [26]. Additionally, machine learning applications have been used to analyze health signals and optimize colony management based on predictive

patterns [27]. Environmental data streams, including microclimate, hive weight, and diurnal cycles, are now routinely collected to inform health assessments and enhance productivity [28].

Swarm detection and motion analysis have also evolved through radar-based systems capable of capturing flight frequency and trajectory with high precision [29]. Computer vision applications, particularly those using YOLO-based object detection, offer automated monitoring of colony entrances, detecting forager numbers, pollen load, and potential threats such as varroa mites [30]. In Brazil, stingless bee monitoring has been strongly aligned with ecological preservation and economic sustainability, driving the integration of IoT infrastructure with policy support [2]. Simultaneously, lightweight and modular systems, often relying on open-source hardware such as Arduino and ESP8266, continue to play a central role in decentralized deployments [31].

Further efforts have RFID-based data collection for evaluating pesticide exposure, which remains one of the most pressing threats to pollinator viability [32]. Policy recommendations increasingly emphasize the need for regulatory frameworks that ensure sustainable beekeeping practices while supporting the adoption of digital technologies [33]. In this context, wireless systems for tracking hive parameters, such as temperature and honey yield, are proving vital to optimizing production strategies [17],[22]. RFID also supports individual bee tracking, facilitating more nuanced behavioral research [34].

Integrated systems such as HiveLink [35] and IntelliBeeHive [36] illustrate how visual data, sensor readings, and AI analytics can converge to offer non-invasive, real-time diagnostic support. These systems demonstrate how timely insights can reduce colony loss and improve beekeeper response. Complementing this direction, Liang [37] proposed a hybrid vision-audio health evaluation model that achieved over 92% classification accuracy, showing particular promise in low-light or acoustically active environments.

Recent developments emphasize not only precision but also environmental breadth. Real-time monitoring solutions now include air quality indices such as CO_2, NH_3, and NOx concentrations measured through LoRa networks [38]. The iBees system, for instance, integrates GPS, thermal, and humidity tracking with cloud-based storage for production optimization [39]. Rosli et al. [40] further expanded this design with the addition of weighing and pressure sensors for hive condition profiling. These solutions highlight the continuous evolution of intelligent beekeeping, reflecting growing demands for accuracy, usability, and adaptability in diagnostic system design.

Finally, diagnostic expert systems like MoViCES, paired with mobile diagnostic tools such as Dr. Kelulut, reflect a broader shift toward accessible, user-centric tools built on the Model-View-Controller paradigm [41]. These platforms enable real-time hive inspection using embedded borescopes and interface logic designed to support non-expert users. Together, these technological pathways form the basis for next-generation diagnostic systems aimed at securing stingless bee populations through adaptive, evidence-driven monitoring.

III. Proposed Initial Model

Compared to existing stingless bee monitoring approaches, which typically focus on either environmental sensing or activity observation in isolation, this study proposes an integrated diagnostic model that combines environmental parameters (temperature and humidity) with behavioral indicators (bee entry/exit frequency). This dual-source data approach enables earlier and more reliable detection of colony stress. Additionally, the classification thresholds are not arbitrarily set; they are refined through expert consultation, ensuring that the system aligns with practical field conditions. The architecture is designed with modularity and scalability in mind, allowing the integration of additional sensors in future iterations. This combination of environmental–behavioral integration, expert-validated thresholds, and expandable architecture addresses key limitations in existing systems and enhances applicability for real-world meliponiculture.

The environmental parameters, such as temperature and humidity, are widely acknowledged as primary indicators of hive conditions. In early-stage research and prototype planning, these two variables were selected as the core inputs for the proposed diagnostic framework. The rationale behind this choice was twofold: first, temperature and humidity sensors are readily available at low cost, and second, these environmental metrics directly influence key biological processes within the stingless bee colony, including brood development, foraging efficiency, and thermoregulation.

In addition to monitoring environmental parameters, the proposed system integrates a behavioral-monitoring module to quantify colony activity. A video camera is placed at the hive entrance to capture continuous footage of bee traffic. Recorded video is processed with a YOLO-based object detection algorithm to identify and count individual bees per frame. These counts are aggregated over time to generate an activity index that reflects foraging and hive-traffic dynamics. Significant deviations from the colony's baseline activity, such as sudden drops in bee traffic during normally busy foraging periods, can signal environmental stressors, disease onset, or other disruptions. By integrating this behavioral stream with environmental temperature and humidity data, the system achieves enhanced sensitivity in detecting early-warning signs that may not manifest in environmental parameters alone [42],[43],[44],[45].

The conceptual model, as shown in Fig. 1, hereafter referred to as the Proposal Model, was designed to accept temperature and humidity readings, compare them against defined thresholds, and generate a basic health classification for the colony. The output is presented in one of three health categories: Healthy, Medium-Healthy, and Least-Healthy, each represented by a range of percentage scores (71 to 100%, 30 to 70%, and 0 to 29%, respectively). This scoring range is intended to give beekeepers an intuitive understanding of their

colony's status, without requiring specialized knowledge to interpret raw sensor data.

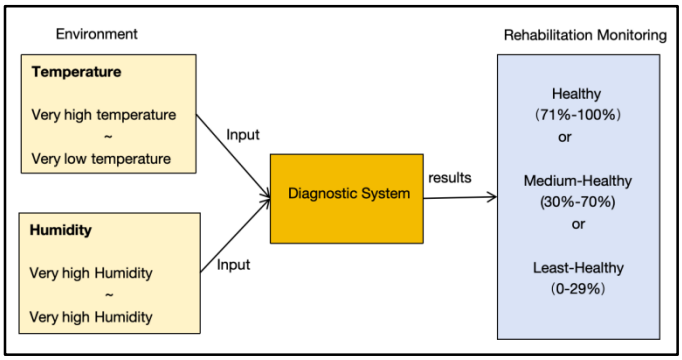


Fig. 1. The conceptual model.

The system begins with environmental sensing units placed within or adjacent to the hive. Two major environmental factors are processed:

Temperature: measurements beyond 25℃ to 34℃ are not optimal. Temperatures over 35℃ may lead to overheating of the brood and a reduction in the activity of worker bees. On the other hand, slowing metabolism and foraging are risks at temperatures below 20℃ [46].

Humidity: fostering fungal growth may occur with excessive humidity over 75%, whereas below 50% contributes to desiccation of the brood and pollen stores [47].

The three health categories: Healthy (71 to 100%), Medium-Healthy (30 to 70%), and Least-Healthy (0 to 29%), were calibrated based on optimal temperature and humidity ranges for stingless bee colonies reported in previous studies. Published data indicate that brood development and colony stability are best maintained within 25 to 34 ℃ and 60 to 85% relative humidity. These threshold values were further refined through consultation with six domain experts, incorporating observational data from two local meliponiculture sites. The combination of literature-based ranges and expert validation ensures that the classification rules reflect both empirical evidence and local field conditions, thereby improving diagnostic reliability and reducing false alerts.

Once inputs are captured, the model uses a rule-based approach to capture deviation from the optimal range. Each input is assessed individually. In the case that both parameters reside within the defined safe zone, the model outputs a Healthy classification. If one variable deviates while the other is acceptable, the system designates the colony as Medium-Healthy. If both exceed warning thresholds, the system marks the colony as Least-Healthy, indicating it is at risk and requires closer examination.

This system makes it possible to track and analyze health status over time. Data from multiple time points can be collected to map out and analyze defined changes over time. For example, persistent classification as Medium-Healthy signals potential unaddressed stress that, if unaddressed, could result in sudden collapse.

Even though the Proposal Model is easy to use and understand, it has some limitations. It considers the relationship between the environment and the colony as fixed and therefore oversimplifies complex systems. It does not account for minor shifts in temperature or even the tiny changes in the environment that bees respond to, such as clustering or fanning. Moreover, environmental parameters do not sufficiently depict the internal processes of the colony, especially in the case of behavioral cues signaling distress long before temperature or humidity changes can be detected.

Moreover, external environmental conditions could alter humidity and temperature readings temporarily, which might activate distress signals. This reinforces the need for a system that goes beyond environmental inputs and considers the cross-validation of the signals with behavioral indicators, such as activity and resting rhythms, foraging habits, and motion density around hive entrances.

Understanding these limitations, the next section incorporates expert feedback to explain how precisely their input was used to improve the model. These insights induced a number of changes that resulted in the creation of the more thorough behavior-inclusive, diagnostic model framework.

IV. Expert Review and Model Refinement

A. Purpose of Expert Review

Prior to field deployment, it is essential for diagnostic models to undergo validation not only through technical testing but also via theoretical scrutiny by domain experts. The purpose of conducting an expert review at this development stage is to evaluate the conceptual soundness, structural feasibility, and practical applicability of the proposed environmental diagnostic model. Expert feedback helps uncover blind spots, refine assumptions, and ensure that system design aligns with the operational context of stingless bee rehabilitation.

B. Expert Panel and Review Scope

A panel of six domain experts was assembled, selected based on their academic and applied experience in the fields of meliponiculture, environmental monitoring, and system modeling. Each expert was invited to participate in a structured review process using a detailed questionnaire.

The scope of the review encompassed four functional aspects of the model: the conceptual framework linking environmental variables to colony health, the diagnostic module responsible for classification based on sensor data, the behavioral monitoring module enhanced by object detection, and the system architecture including modularity, scalability, and usability.

C. Review of Instrument Design

The expert review was conducted using a structured questionnaire designed to support both quantitative assessment and qualitative feedback. The instrument was organized into four sections, aligned with the functional modules of the model. Each section included closed-ended items on a three-point Likert scale: "Agree, no modification", "Agree, with

modification", and "Disagree", as well as open-text fields to capture elaborative suggestions.

The questionnaire design was informed by validation practices in information systems and agricultural technology research. Particular attention was paid to clarity, neutral phrasing, and accessibility across disciplinary backgrounds.

1) Conceptual framework: Evaluates the theoretical basis of the model, particularly its capacity to represent the relationship between environmental stressors (temperature and humidity) and colony health. Items assess whether the model adequately reflects the real-world dynamics of stingless bee rehabilitation.

2) Diagnostic module: Assesses the model environmental sensing component, including the use of DHT22 sensors, the design of health classification thresholds, and the practicality of data capture methods in field conditions.

3) Behavioral monitoring module: Focuses on the integration of computer vision, particularly YOLO-based activity detection, and its relevance for monitoring early indicators of colony stress.

4) System architecture: Evaluates the structure of the system from the perspective of modularity, upgradeability, and end-user usability, including questions related to interface logic, data separation, and adaptability to new sensors.

The instrument emphasized ethical standards, ensuring expert anonymity, voluntary participation, and informed consent.

D. Evaluation Criteria

The evaluation criteria were structured to assess the scientific rigor, operational viability, and ecological relevance of the model. These included:

1) Accuracy of environmental variable selection: Whether temperature and humidity sufficiently capture key influences on stingless bee colony health.

2) Diagnostic reliability: Ability of the classification module to process sensor data, apply thresholds, and produce actionable health scores.

3) Behavioral analysis validity: Effectiveness of the activity detection scheme and its adaptability to field conditions.

4) System modularity and scalability: Capacity for future extension, and separation of functional layers (data capture, processing, interface).

Experts were invited to provide improvement suggestions, highlight oversights, and recommend additional system capabilities where applicable.

E. Questionnaire Administration and Data Collection

Questionnaires were distributed electronically to the six selected experts. Each received a complete packet including a system overview, visual schematics of the model, and an explanation of the review purpose.

The structured responses were compiled and analyzed thematically. Results indicated strong agreement on the model conceptual clarity, with minor modifications suggested for threshold ranges and behavior classification granularity.

Insights gained from this review directly informed the revision of the improved model, ensuring that the final prototype is both scientifically grounded and practically deployable in real-world meliponiculture contexts.

V. Results and Analysis of Expert Feedback

Gaining feedback through the expert review process revealed both the strengths and weaknesses of the environmental diagnostic model. The systematic evaluation and open-ended commentary provided by six experts in the domain allowed for the appraisal of the model integrity, feasibility, and technical scope. Their responses were crucial to the refinement process of the model which took place prior to prototyping and implementation.

In summary, specialized consensus recognized that temperature and humidity are the crucial parameters for the well-being of stingless bee colonies. Focus group members argued that there exists sufficient literature and empirical evidence—both for and field data—to support these factors as significant for brood survival and foraging activity. While some specialists proposed to study additional factors like pesticide, rainfall, or wind, most respondents believed that these inclusions would be too complex for the model, adding unnecessary system complexity and making data collection impractical. Therefore, the model was considered to have an adequately narrow scope for its initial phase while still allowing for these additional factors to be included in later revisions.

Reviewers concentrated on the empirical thresholds as the classification logic and stressed the necessity to define these values more rigorously. These critiques were addressed in the revised model which incorporated predefined empirical ranges for temperature and humidity based on field study data and data collected at the sites. This change was made to improve the objectivity and transparency of the classification procedure, especially in the differentiation of Healthy, Medium Healthy, and Least Healthy. Besides the fixed boundaries, the precise measurement and observation nexus was simplified to integrate more behavioral input to strengthen classification consistency.

The incorporation of computer vision to monitor activity patterns was positively received. Experts agreed that bee behavior often reflects underlying stress before environmental parameters shift dramatically. The use of YOLO-based movement tracking was considered a practical and innovative approach, provided lighting conditions and camera angles are well-controlled. Some experts encouraged exploring audio analysis in future model iterations, especially for applications in low-light environments or enclosed hives where visual data may be unreliable.

Further feedback addressed the overall system architecture, particularly the modularity and potential for scalability. Experts viewed the separation of data collection, processing, and interface layers as beneficial for future upgrades. The improved model maintains compatibility with additional sensors and

analytics modules, allowing the system to evolve with user needs and technological advancements.

The revised diagnostic model, presented as Fig. 2, reflects the collective input received through the expert review process. The visual representation now illustrates the integration of clearly defined environmental thresholds, behavior scoring modules, and a composite health assessment logic. While some expert recommendations—such as multi-factor environmental correlation—were deemed outside the current scope due to limited data availability, the model retains flexibility for phased development.

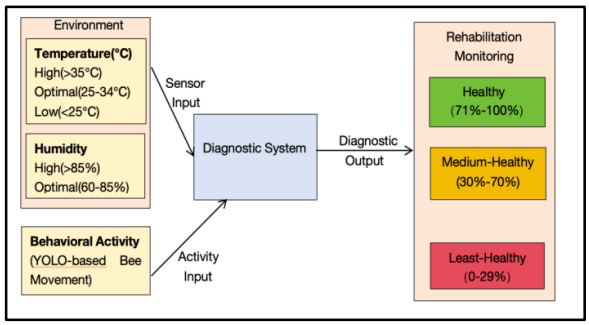


Fig. 2. The improved diagnostic model.

A total of six domain experts participated in the validation of the proposed environmental diagnostic model. The evaluation combined structured questionnaire items with open-ended feedback to capture both quantitative agreement levels and qualitative improvement suggestions.

Regarding environmental parameter selection, five experts (83%) fully agreed that temperature and humidity are the most critical indicators for stingless bee colony health, supported by literature and field experience. One expert (17%) recommended the inclusion of additional variables such as pesticide exposure, rainfall, and wind speed for future model iterations.

For threshold definitions, the original model expressed temperature and humidity using broad qualitative categories (ranging from "very high" to "very low"). Based on expert feedback and literature review, these were replaced with quantitative ranges. The temperature threshold was refined to 25℃ to 34℃, aligning with documented optimal conditions for brood development and adult bee activity, and narrowing the previous qualitative extremes to improve measurement precision. The humidity threshold was set to 60% to 85%, representing a narrowing from the broader original qualitative range. This adjustment reduces classification ambiguity and better reflects the optimal moisture conditions for maintaining brood health, as confirmed by preliminary site measurements.

In terms of behavioral monitoring, all six experts (100%) endorsed the integration of computer vision for activity tracking. Four (67%) supported the use of entry/exit frequency and foraging intensity as primary indicators, while two (33%) proposed adding in-nest movement metrics for more granular behavioral assessment. Future work will investigate the feasibility of incorporating these additional in-nest movement metrics, as they could provide deeper insights into colony therm.

On system architecture, five experts (83%) agreed that the modular design separating data collection, processing, and user interface layers supports scalability and future integration of additional sensors or analytics modules. One expert (17%) highlighted the need for detailed user interface customization to match local beekeeper practices. To address this, future work will include developing a customizable user interface framework that allows adaptation to different cultural, linguistic, and operational contexts, thereby enhancing the model's usability and facilitating its broader adoption.

Overall, the statistical results indicate strong expert consensus on the core environmental and behavioral parameters, with constructive recommendations aimed at refining threshold precision and expanding the behavioral dataset. The incorporation of these validated ranges and prioritized behavioral metrics into the improved model enhances both its scientific robustness and its practical applicability in real-world beekeeping contexts.

By aligning expert opinion with empirical research and real-world constraints, the refinement process has strengthened both the diagnostic logic and the system relevance to field practitioners. The structured feedback not only confirmed the conceptual robustness of the model but also identified practical adjustments that improve usability, interpretability, and future integration potential. These insights will guide the next stage of model validation and system deployment.

VI. Discussion

The integration of environmental parameters with bee activity data provides a holistic perspective on colony health, addressing limitations in traditional monitoring approaches that rely solely on manual observation or single data types. Expert feedback further supports the model's practical feasibility, highlighting its potential to inform proactive interventions. However, the current threshold definitions remain in a preliminary stage, as they were derived from a combination of literature-based ranges and short-term data from a limited number of colonies under specific seasonal and geographic conditions. This restricted dataset may not capture variability due to climate, floral availability, or hive management practices in other regions.

To achieve more definitive calibration, future work should follow a structured roadway: 1) conduct longitudinal monitoring across multiple colonies, seasons, and habitat types; 2) integrate external environmental datasets (local meteorological data) for contextual correlation; 3) perform statistical sensitivity analyses to determine optimal threshold boundaries; and 4) validate the refined model through cross-site field trials with independent beekeeper cohorts. This multi-stage approach will enable the thresholds to be both scientifically robust and broadly applicable to diverse meliponiculture contexts.

Compared to existing stingless bee monitoring systems that rely solely on either environmental sensors or behavioral observations, the proposed model delivers a more comprehensive health assessment by integrating both data sources into a unified diagnostic framework. The inclusion of expert-validated thresholds ensures that the classification logic

is both scientifically grounded and field-ready. Furthermore, the modular architecture allows for future upgrades, such as adding sound analysis, air quality sensing, or advanced behavioral metrics, without requiring major system redesign. This combination of data integration, expert refinement, and practical scalability enhances the model's originality and applicability for sustainable stingless bee colony management [46], [48].

VII. Conclusion

This study has presented the development and refinement of an environmental diagnostic model tailored to the health monitoring needs of stingless bee colonies. Recognizing the limitations of conventional inspection methods and single-parameter sensing systems, the model integrates both environmental variables namely temperature and humidity and behavioral indicators derived from computer vision analysis to generate a composite health assessment. This multidimensional framework enables earlier detection of colony stress and delivers actionable insights in a format suitable for field practitioners.

To ensure conceptual clarity and practical relevance, the model underwent a structured expert review involving six specialists across meliponiculture, environmental sensing, and system design. Their feedback validated the scientific basis of the model, affirmed the choice of key parameters, and highlighted important refinements, such as clearer threshold definitions and improved interpretability. As a result, the revised model incorporates empirically grounded environmental ranges, activity-based behavioral detection, and a modular architecture that supports future system enhancement.

Future work will focus on deploying the system in live apiary environments to evaluate real-time performance, robustness under variable weather conditions, and end-user interaction feedback. In parallel, work will continue on refining the behavioral analysis component, exploring the use of hybrid vision and sound recognition to enhance accuracy in low-visibility settings.

In conclusion, this expert-informed environmental diagnostic model represents a promising step towards more intelligent, interpretable, and scalable monitoring tools for stingless bee colony rehabilitation. By combining ecological understanding with low-cost sensing and expert knowledge, it contributes not only to applied beekeeping technologies, but also to the broader goal of sustaining pollinator health in increasingly stressed environments.

Acknowledgment

The authors gratefully acknowledge the guidance and support of Dr. Rozi Nor Haizan Nor throughout the development of this study. The authors also extend sincere thanks to the six domain experts who contributed their time and insights through the expert questionnaire, which significantly informed the refinement of the proposed model.

References

[1] Toledo-Hernández, E., Peña-Chora, G., Hernandez-Velazquez, V. M., Lormendez, C. C., Toribio-Jiménez, J., Romero-Ramírez, Y., & León-Rodríguez, R. (2022). The stingless bees (Hymenoptera: Apidae: Meliponini): A review of the current threats to their survival. Apidologie, 53 (1), 8.

[2] Jaffé, R., Pope, N., Carvalho, A. T., Maia, U. M., Blochtein, B., de Carvalho, C. A. L., Carvalho-Zilse, G. A., Freitas, B. M., Menezes, C., Ribeiro, M. de F., Venturieri, G. C., & Imperatriz-Fonseca, V. L. (2015). Bees for development: Brazilian survey reveals how to optimize stingless beekeeping. PLoS ONE, 10 (3), e0121157. https://doi.org/10.1371/journal.pone.0121157.

[3] Zapata-Hernández, G., Gajardo-Rojas, M., Calderón-Seguel, M., Muñoz, A. A., Yáñez, K. P., Requier, F., Fontúrbel, F. E., Ormeño-Arriagada, P. I., & Arrieta, H. (2024). Advances and knowledge gaps on climate change impacts on honey bees and beekeeping: A systematic review. Global Change Biology, *30*(3), e17219. https://doi.org/10.1111/gcb.17219.

[4] Danieli, P. P., Addeo, N. F., Lazzari, F., Manganello, F., & Bovera, F. (2023). Precision beekeeping systems: State of the art, pros and cons, and their application as tools for advancing the beekeeping sector. Animals, 14(1), 70. https://doi.org/10.3390/ani14010070.

[5] Minaud, É., Rebaudo, F., Davidson, P., Hatjina, F., Hotho, A., Mainardi, G., Steffan-Dewenter, I., Vardakas, P., Verrier, E., & Requier, F. (2024). How stressors disrupt honey bee biological traits and overwintering mechanisms. Heliyon, 10, e34390. https://doi.org/10.1016/j.heliyon.2024.e34390.

[6] Astuti, P. K., Hegedűs, B., Oleksa, A., Bagi, Z., & Kusza, S. (2024). Buzzing with intelligence: Current issues in apiculture and the role of artificial intelligence (AI) to tackle it. Insects, 15(6), 418. https://doi.org/10.3390/insects15060418.

[7] Kontogiannis, S., et al. (2024). Beehive smart detector device for the detection of critical conditions that utilize edge device computations and deep learning inferences. Sensors, 24(16), 5444. https://doi.org/10.3390/s24165444.

[8] Alleri, M., Amoroso, S., Catania, P., Lo Verde, G., Orlando, S., Ragusa, E., Sinacori, M., Vallone, M., & Vella, A. (2023). Recent developments on precision beekeeping: A systematic literature review. Journal of Agriculture and Food Research, 14, 100726. https://doi.org/10.1016/j.jafr.2023.100726.

[9] Williams, S. M. (2024). Tracking and predicting bee behaviour and movement using machine learning [Doctoral dissertation, Bangor University (United Kingdom)].

[10] Kulyukin, V. A., Coster, D., Kulyukin, A. V., Meikle, W., & Weiss, M. (2024). Discrete time series forecasting of hive weight, in-hive temperature, and hive entrance traffic in non-invasive monitoring of managed honey bee colonies: part I. Sensors, 24(19), 6433.

[11] Arendela, R. J., Ebora, R. A., Arboleda, E., Ramos, J. L. M., Bono, M., & Dimero, D. (2024). IoT stingless bee colony monitoring system. Journal of Agricultural Machinery, 2024. https://doi.org/10.22067/jam.2024.86261.1222.

[12] Månefjord, H., Huzortey, A. A., Boateng, R., Gbogbo, Y. A., Yamoa, A. S. D., Zoueu, J. T., Kwapong, P. K., Anderson, B., & Brydegaard, M. (2024). Hyperspectral lidar for monitoring high-resolution activity patterns of African stingless bee species. Animal Biotelemetry, 12, Article 15. https://doi.org/10.1186/s40317-024-00372-3.

[13] Peñaloza-Aponte, D., Brandt, S., Dent, E., Underwood, R. M., DeMoras, B., Bruckner, S., López-Uribe, M. M., & Urbina, J. V. (2024). Automated entrance monitoring to investigate honey-bee foraging trips using an open-source wireless platform and fiducial tags. HardwareX, 20, e00609. https://doi.org/10.1016/j.ohx.2024.e00609.

[14] Ahmad, S., & Colleagues. (2024). Machine-learning based acoustic analysis of stingless bee (Heterotrigona itama) alarm signals during intruder events. Agriculture, 15(6), 591. https://doi.org/10.3390/agriculture15060591.

[15] Rosa, A. de S., Price, R. I., Caliman, M. J. F., Queiroz, E. P., Blochtein, B., Pires, C. S. S., & Imperatriz-Fonseca, V. L. (2015). The stingless bee species, Scaptotrigona aff. depilis, as a potential indicator of environmental pesticide contamination. Environmental Toxicology and Chemistry, 34(8), 1851–1853. https://doi.org/10.1002/etc.3003.

[16] Abou-Shaara, H. F., Owayss, A. A., Ibrahim, Y. Y., & Basuny, N. K. (2017). A review of impacts of temperature and relative humidity on

various activities of honey bees. Insectes Sociaux, 64, 455–463. https://doi.org/10.1007/s00040-017-0573-8.

[17] Cecchi, S., Spinsante, S., Terenzi, A., & Orcioni, S. (2020). A smart sensor-based measurement system for advanced bee hive monitoring. Sensors, 20 (9), 2726. https://doi.org/10.3390/s20092726.

[18] Jeong, K., Oh, H., Lee, Y., Seo, H., Jo, G., & Jeong, J. (2024). IoT and AI Systems for Enhancing Bee Colony Strength in Precision Beekeeping: A Survey and Future Research Directions. IEEE Internet of Things Journal. https://doi.org/10.1109/JIOT.2024.3461775.

[19] König, A. (2019). IndusBee 4.0–integrated intelligent sensory systems for advanced bee hive instrumentation and hive keepers' assistance systems. Sensors & Transducers, 237 (9/10), 109–121.

[20] Russell, S., Barron, A. B., & Harris, D. (2013). Dynamic modelling of honey bee (Apis mellifera) colony growth and failure. Ecological Modelling, 265, 158–169. https://doi.org/10.1016/j.ecolmodel.2013.06.005.

[21] Miskon, M. T., Abdul Razak, M. A., Jaafar, H., Hassan, N., & Mahmood, R. (2022). A Lora-based Testbed Development for Stingless Bee Monitoring System. TEM Journal, 11 (2), 712–719. https://doi.org/10.18421/TEM112-17.

[22] Zaman, A., & Dorin, A. (2023). A framework for better sensor-based beehive health monitoring. Computers and Electronics in Agriculture, 210, 107906. https://doi.org/10.1016/j.compag.2023.107906.

[23] Radaeski, J. N., & Bauermann, S. G. (2021). Contributions to melissopalynology studies in southern Brazil: Pollen analysis in the honeys from Apis mellifera, Tetragonisca angustula, Melipona quadrifasciata quadrifasciata, Scaptotrigona bipunctata, Plebeia remota and Plebeia droryana. Palynology, 45 (3), 477–486. https://doi.org/10.1080/01916122.2021.1878307.

[24] Becher, M. A., Osborne, J. L., Thorbek, P., Kennedy, P. J., & Grimm, V. (2013). Towards a systems approach for understanding honeybee decline: A stocktaking and synthesis of existing models. Journal of Applied Ecology, 50 (4), 868–880. https://doi.org/10.1111/1365-2664.12112.

[25] Anuar, N. H. K., Yunus, M. A. M., Baharudin, M. A., Ibrahim, S., Sahlan, S., Hashem, M. Y. B., & Siddique, M. A. (2023, May). Data management system of IoT for stingless bee monitoring. In AIP Conference Proceedings (Vol. 2795, No. 1). AIP Publishing. https://doi.org/10.1063/5.0156870.

[26] Imoize, A. L., Odeyemi, S. D., & Adebisi, J. A. (2020). Development of a low-cost wireless bee-hive temperature and sound monitoring system. Indonesian Journal of Electrical Engineering and Informatics, 8(3), 476–485. https://doi.org/10.52549/ijeei.v8i3.2268.

[27] Ali, M. C., Ilias, B., Rahim, N. A., Shukor, S. A., Adom, A. H., & Saad, M. A. H. (2021, November). A review on the stingless beehive conditions and parameters monitoring using IoT and machine learning. In Journal of Physics: Conference Series (Vol. 2107, No. 1, Article 012040). IOP Publishing. https://doi.org/10.1088/1742-6596/2107/1/012040.

[28] Braga, A. R., Gomes, D. G., Rogers, R., Hassler, E. E., Freitas, B. M., & Cazier, J. A. (2020). A method for mining combined data from in-hive sensors, weather and apiary inspections to forecast the health status of honey bee colonies. Computers and Electronics in Agriculture, 169, 105161. https://doi.org/10.1016/j.compag.2019.105161.

[29] Aumann, H. M., Aumann, M. K., & Emanetoglu, N. W. (2021). Janus: A combined radar and vibration sensor for beehive monitoring. IEEE Sensors Letters, 5 (3), 1-4. https://doi.org/10.1109/LSENS.2021.3061490.

[30] Bilik, S., Kratochvila, L., Ligocki, A., Bostik, O., Zemcik, T., Hybl, M., Horak, K., & Zalud, L. (2021). Visual diagnosis of the Varroa destructor parasitic mite in honeybees using object detector techniques. Sensors, 21 (8), 2764. https://doi.org/10.3390/s21082764.

[31] Rosli, H. A., Abdul Malik, N., & Ahmad, Y. A. (2022). IoT Based Monitoring System for Stingless Bees Colony in IIUM. Journal of Physics: Conference Series, 2312 (1), 012088. https://doi.org/10.1088/1742-6596/2312/1/012088.

[32] Schneider, C. W., Tautz, J., Grünewald, B., & Fuchs, S. (2012). RFID tracking of sublethal effects of two neonicotinoid insecticides on the foraging behavior of Apis mellifera. PLoS ONE, 7 (1), e30023. https://doi.org/10.1371/journal.pone.0030023.

[33] Ismail, W. I. W. (2016). A review on beekeeping in Malaysia: History, importance, and future directions. Journal of Sustainable Science and Management, 11 (1), 70–80.

[34] Nunes-Silva, P., Hrncir, M., Guimarães, J. T. F., Arruda, H., Costa, L., Pessin, G., Siqueira, J. O., de Souza, P., & Imperatriz-Fonseca, V. L. (2019). Applications of RFID technology on the study of bees. Insectes Sociaux, 66(1), 15–24. https://doi.org/10.1007/s00040-018-0661-4.

[35] Dsouza, A., P, A., & Hegde, S. (2023). HiveLink, an IoT based Smart Bee Hive Monitoring System. arXiv preprint arXiv:2309.12054. https://arxiv.org/abs/2309.12054.

[36] Narcia-Macias, C. I., Guardado, J., Rodriguez, J., Park, J., Rampersad-Ammons, J., & Enriquez, E. (2023). IntelliBeeHive: An Automated Honey Bee, Pollen, and Varroa Destructor Monitoring System. arXiv preprint arXiv:2309.08955. https://arxiv.org/abs/2309.08955.

[37] Liang, A. (2024). Developing an AI-based Integrated System for Bee Health Evaluation. IEEE Access, 12, 70025–70038. https://doi.org/10.1109/ACCESS.2024.3464559.

[38] Ramli, M. F., Harun, A., Ismail, R. C., Ndzi, D. L., Turner, J. S. C., Farook, R. S. M., Isa, M. N. M., & Murad, S. A. Z. (2021). Design and development of stingless beehive air pollutant monitoring system. International Journal of Nanoelectronics and Materials, 14 (Special Issue), 185–196.

[39] Man, M., Wan Abu Bakar, W. A., & Abdul Razak, M. A. B. B. (2019). An intelligent stingless bee system with embedded IOT technology. International Journal of Recent Technology and Engineering (IJRTE), 8 (3), 264–269. https://doi.org/10.35940/ijrte.C3985.098319.

[40] Rosli, H. A., Abdul Malik, N., & Ahmad, Y. A. (2022). IoT Based Monitoring System for Stingless Bees Colony in IIUM. Journal of Physics: Conference Series, 2312 (1), 012088. https://doi.org/10.1088/1742-6596/2312/1/012088.

[41] Izzatul Lail, M. Y. (2021). Movices architecture for Meliponiculture Diagnostic Application System Development [Master's thesis, Universiti Putra Malaysia]. p. 18.

[42] Kulyukin, V. A., & Kulyukin, A. V. (2023). Accuracy vs. Energy: An Assessment of Bee Object Inference in Videos from On-Hive Video Loggers with YOLOv3, YOLOv4-Tiny, and YOLOv7-Tiny. Sensors, 23(15), 6791. https://doi.org/10.3390/s23156791.

[43] Lei, C., Lu, Y., Xing, Z., Zhang, J., Li, S., Wu, W., & Liu, S. (2024). A Honey Bee In-and-Out Counting Method Based on Multiple Object Tracking Algorithm. Insects, 15(12), 974. https://doi.org/10.3390/insects15120974.

[44] Narcia-Macias, C. I., Guardado, J., Rodriguez, J., Rampersad-Ammons, J., Enriquez, E., & Kim, D.-C. (2023). IntelliBeeHive: An Automated Honey Bee, Pollen, and Varroa Destructor Monitoring System (preprint). arXiv. https://doi.org/10.48550/arXiv.2309.08955.

[45] Sledevič, T., Serackis, A., Matuzevičius, D., Plonis, D., & Vdoviak, G. (2025). Visual recognition of honeybee behavior patterns at the hive entrance. PLoS ONE, 20(2), e0318401. https://doi.org/10.1371/journal.pone.0318401.

[46] Vollet-Neto, A., Menezes, C., & Imperatriz-Fonseca, V. L. (2015). Behavioural and developmental responses of a stingless bee (Scaptotrigona depilis) to nest overheating. Apidologie, *46*, 455–464. https://doi.org/10.1007/s13592-014-0336-8.

[47] Maia-Silva, C., Imperatriz-Fonseca, V. L., Silva, C. I., & Hrncir, M. (2014). Environmental windows for foraging activity in stingless bees, Melipona subnitida Ducke and Melipona quadrifasciata Lepeletier (Hymenoptera: Apidae: Meliponini). Sociobiology, *61*(4), 378–385. https://doi.org/10.13102/sociobiology.v61i4.378-385.

[48] Arendela, R. J., Ebora, R. A., Arboleda, E., Ramos, J. L. M., Bono, M., & Dimero, D. (2025). IoT Stingless Bee Colony Monitoring System. Journal of Agricultural Machinery, 15(1), 47–63. https://doi.org/10.22067/jam.2024.86261.1222.

A Secure Authentication Protocol for IoT Devices

Mohamed Ech-Chebaby, Hicham Zougagh, Hamid Garmani, Zouhair Elhadari
Faculty of Sciences and Techniques, Moulay Slimane University, Beni Mellal, Morocco

***Abstract*—The rapid evolution of the Internet of Things (IoT) offers vast opportunities in automation and connectivity, yet simultaneously introduces critical security challenges. One of the most pressing concerns lies in the heterogeneity and limited computational capabilities of IoT devices, which complicate the deployment of robust security mechanisms. In this work, we present a lightweight and secure authentication protocol designed to establish mutual authentication between a server and smart objects. Our protocol enhances the scheme proposed by Fatma et al., addressing its identified vulnerabilities. Formal security analysis using AVISPA and ProVerif confirms the protocol's resilience against a wide range of threats. Furthermore, a practical simulation was conducted using a Raspberry Pi as the IoT device and a Core i5-based server to evaluate real-world performance. Results show that the protocol executes efficiently in real-time with a reduced authentication delay, demonstrating its feasibility for resource-constrained environments. This research contributes to the development of effective, scalable, and secure authentication solutions tailored for the IoT landscape.**

***Keywords*—IoT; Internet of Things; security; authentication**

I. Introduction

The widespread adoption of the Internet of Things (IoT) has transformed the way devices interact and exchange information, leading to a substantial increase in the number of connected smart devices worldwide. This unprecedented growth has amplified the demand for robust security mechanisms, particularly for mutual authentication between devices and servers. In such distributed environments, it is essential to guarantee not only reliable communication but also the confidentiality, integrity, and authenticity of the exchanged data [1], [2], [3].

Authentication is a cornerstone of IoT security, as it establishes trust between communicating entities before any sensitive information is exchanged. Over the years, numerous authentication protocols have been developed to secure the establishment of sessions between servers and smart objects. However, extensive analysis of existing schemes has revealed that many of them remain vulnerable to critical threats such as impersonation, replay, and man-in-the-middle (MitM) attacks. These weaknesses can be exploited by malicious actors to compromise both the confidentiality and integrity of IoT communications, thereby undermining the overall security of the system [4], [5].

In particular, the protocol proposed by Fatma et al. [6], while offering lightweight computation, exhibits several vulnerabilities that may lead to the disclosure of sensitive credentials and session keys. Such flaws pose serious risks in IoT scenarios, especially in resource-constrained environments where devices cannot easily implement computationally intensive countermeasures.

To address these limitations, this study introduces an enhanced authentication protocol designed for IoT environments. Our scheme leverages elliptic curve cryptography (ECC) for efficient key exchange and collision-resistant hash functions for identity protection and message integrity. ECC offers a favorable trade-off between computational efficiency and cryptographic strength, making it especially suitable for devices with constrained processing power, memory, and energy resources. The proposed design incorporates random nonces, key-derived masking, and challenge-response mechanisms to ensure resilience against known attack vectors while maintaining low overhead.

The security of the proposed protocol has been rigorously evaluated using formal verification methods to identify and mitigate potential vulnerabilities before deployment. Simulation and analysis demonstrate that our protocol achieves the following key security properties:

- Mutual authentication between server and device.
- Resistance to impersonation, replay, and MitM attacks.
- Confidentiality of transmitted data and integrity of messages.
- Low computational and communication cost suitable for IoT devices.

By integrating these measures, the proposed protocol provides a significant improvement in both security and operational efficiency over existing schemes. Ultimately, this work aims to contribute to the development of practical, scalable, and secure authentication solutions for the IoT ecosystem, ensuring trustworthy communication even in highly resource-constrained environments.

II. The Mechanisms used by Authentication Protocols

A. Elliptic Curve Cryptography (ECC)

Elliptic Curve Cryptography (ECC) is a well-established cryptographic method known for providing robust security in authentication schemes [7]. It is based on the mathematical structure of elliptic curves, typically represented as:

$$y^2 = x^3 + ax + b \tag{1}$$

where, a and b are curve parameters. In ECC, cryptographic operations involve algebraic manipulations on points belonging to the elliptic curve. For example, adding two points P and Q on the curve yields a third point R, while multiplying a point P by a scalar k produces another valid point on the curve [8].

A key advantage of ECC is that it achieves comparable security to other public-key systems with significantly smaller

key sizes. For instance, a 128-bit ECC key can offer a security level equivalent to that of a 1024-bit RSA key. This property reduces both computational overhead and storage requirements, making ECC highly suitable for resource-constrained devices [8].

The security of ECC is rooted in the hardness of the Elliptic Curve Discrete Logarithm Problem (ECDLP), which involves determining the integer k such that $kP = Q$, where P and Q are known points on the curve. Solving this problem is computationally infeasible for appropriately chosen curve parameters and key sizes [9], [10].

Due to its combination of mathematical robustness, efficiency, and reduced key size requirements, ECC is extensively employed in modern authentication protocols to provide secure entity verification and protect the confidentiality of exchanged information [11].

B. Cryptographic Hash Function

A cryptographic hash function is a mathematical algorithm that converts input data of arbitrary length into a fixed-length output, commonly referred to as a hash or message digest [12]. Such functions are fundamental in numerous areas of computer science, particularly for ensuring data security and verifying information integrity, as illustrated in the following Fig. 1.

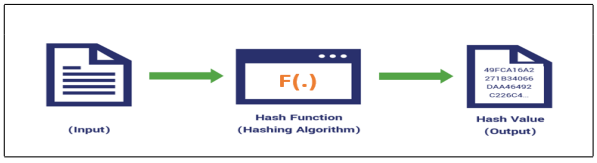


Fig. 1. Cryptographic hash function.

The hash functions possess the following properties:

- Deterministic: A cryptographic hash function is deterministic if the same input consistently yields the same hash value.
- Collision resistance: A hash function has collision resistance when it is computationally infeasible to find two distinct inputs that produce the identical hash output.
- Irreversibility: A hash function is irreversible if recovering the original input from its hash value is computationally impractical or impossible.

The most commonly used cryptographic hash functions include MD5, SHA-1, SHA-256, and SHA-3.

MD5, once widely adopted, is now considered insecure due to identified vulnerabilities and should not be used for security-critical purposes [13].

Similarly, SHA-1, although still present in some systems, is regarded as weak against modern cryptographic attacks [14].

In contrast, SHA-256 and SHA-3 are currently preferred for contemporary applications, as they provide stronger resistance to known attack methods [15].

III. Related Works

Authentication protocols in the Internet of Things (IoT) domain are a frequent subject of discussion in the scientific literature and have recently attracted considerable attention.

In [16], the authors propose a secure multifactor authentication scheme for cloud-based systems, demonstrated in the context of electronic health records. The approach combines three authentication factors—something the user knows, has, and is with cryptographic techniques such as RSA digital signatures and hashed credentials to achieve mutual authentication and ensure data integrity. Key features include an anonymous health center, acting as a trusted third party, responsible for distributing secret keys and managing user permissions during the registration phase. Additionally, the scheme incorporates a QR code mechanism to securely transmit doctor-signed electronic prescriptions to pharmacists. The protocol was implemented with entities such as patients, doctors, and a hospital server, and its security was validated using both the Scyther formal verification tool and informal analysis, demonstrating resistance to man-in-the-middle, replay, impersonation, insider, and phishing attacks.

Boonkrong's work [17] revisits Park et al.'s [18] multifactor biometric remote user authentication scheme and exposes several critical weaknesses, then offers an enhanced protocol to resolve them. Park et al.'s scheme – an improvement over earlier methods – was found to lack adequate message integrity protection (allowing an attacker to modify messages without detection), and freshness guarantees (making replay attacks feasible). It also did not implement a true challenge–response mechanism, leading to incomplete mutual authentication and the possibility of man-in-the-middle attacks, and it provided no way for the client to prove to the server that both share the same session key (leaving session key agreement only partially verified).

In [19], the authors addressed the challenge of safeguarding large volumes of sensitive user data in electronic healthcare (e-health) systems, particularly when using wireless devices with limited processing and storage capabilities. They proposed a dynamic privacy-preserving approach that enables server-side biometric authentication while ensuring complete user anonymity. In their method, the server does not have access to the exact biometric template value, which mitigates the risk of privacy leakage. Moreover, all messages transmitted through their scheme are untraceable, thereby maintaining a high degree of anonymity.

In [20], the researchers focused on the design of an end-to-end mutual authentication protocol for Wearable Health Monitoring Systems (WHMS). They analyzed the mutual authentication protocol proposed by Amin et al. [21] and identified several security flaws. Their improved version, based on quadratic residues, mitigates vulnerabilities such as stolen mobile device attacks, de-synchronization attacks, and sensor key exposure.

The work in [22] addressed the development of lightweight security mechanisms for Wireless Body Area Networks (WBANs), with an emphasis on securely transmitting sensitive patient information. The authors evaluated a lightweight authentication protocol proposed by Liu et al. [23] and, through a security assessment, found weaknesses in its design.

To overcome these limitations, they proposed a single-round lightweight authentication scheme that enhances security and reduces computational overhead.

In [6], an authentication protocol was designed for IoT-based healthcare applications. The authors introduced two major enhancements to the recently proposed M2C mutual authentication protocol, which was originally intended for RFID-based healthcare systems. Their modified protocol, referred to as M2M, leverages Elliptic Curve Cryptography (ECC) to secure RFID communications in healthcare environments.

The authors of [24] proposed an authentication mechanism for Medical Body Area Networks (MBANs) based on patient body motion. They developed a generalized model capable of characterizing routine patient activities—such as walking and running—to verify the legitimacy of sensor nodes. Their security analysis demonstrated that the scheme is resistant to well-known attack vectors.

Using ECC, [25] proposed an RFID-based mutual authentication scheme aimed at improving patient medication safety. The scheme enables secure exchanges between an RFID tag and a medication server, ensuring reliable medical evidence for prescriptions and dosage administration. Nevertheless, this protocol was later shown to have vulnerabilities, which were addressed by Fatma et al. [6].

In [26], the authors presented an authentication protocol for IoT-based RFID systems, focusing on mitigating security risks inherent in wireless communication channels. Their design integrates ECC and employs the elliptic curve Diffie–Hellman (ECDH) [27] key agreement mechanism to generate temporary shared keys for encrypting transmitted messages.

Several studies have also presented mutual authentication schemes aimed at establishing a secure session key between RFID tags and backend servers to ensure reliable communication. Examples include the works of Dinarvand et al. [28], Liao et al. [29] for specialized environments, and Zhao et al. [30] for healthcare-specific applications.

All these schemes generally follow a two-phase structure. The configuration phase involves the server generating ECC public/private key pairs to be used in the subsequent stage. The authentication phase is then repeated each time the server initiates a connection with a smart object, during which both entities exchange messages to mutually verify their identities.

IV. Description and Notation Used

Before presenting our enhanced version of the M2C authentication protocol derived from the scheme proposed by Fatma et al. [6], Table I summarizes the notations employed in its description.

V. Review and Cryptanalysis of Fatma et al.'s Protocol

Fatma et al. [6] introduced an ECC-based authentication scheme tailored for resource-limited devices, including RFID tags and sensors. Owing to its design, it falls under the category of machine-to-cloud (M2C) protocols. The proposed approach functions through two primary phases.

TABLE I. Notations used in the Protocol Description (Modified Symbols)

Notation	Meaning
n_{tot}	Total of IoT devices.
$\mathcal{S}$	Server in Cloud.
$\mathcal{D}$	IoT device.
D_k	Specific IoT device D_k, where $k \in [1, n_{tot}]$
q_a, q_b	Two large prime numbers.
$\mathbb{F}^*_{q_a}$	Multiplicative group of a finite field.
$\mathcal{E}$	Elliptic curve defined by $y^2 = x^3 + ax + b$.
P_g	Base point (generator) of $\mathcal{E}$ of order q_a.
(q_b, a, b, q_a, P_g)	Domain parameters for constructing $\mathcal{E}$.
H	Collision-resistant secure hash function.
$\rho \xleftarrow{\$} \mathbb{F}^*_{q_a}$	Random element ρ sampled from $\mathbb{F}^*_{q_a}$.
$\alpha \cdot \beta$	Scalar multiplication of α with elliptic curve point β.
$\alpha \oplus \beta$	XOR operation.
$\alpha \parallel \beta$	Concatenation operation of α and β.
$\alpha \stackrel{?}{=} \beta$	Equality check.
ID_{D_k}	Unique identity of device D_k.
hID_{D_k}	Hash of the identity ID_{D_k} stored in the server database.
$\mathsf{SK}_{\mathcal{S}}$	Server's private key.
$\mathsf{PK}_{\mathcal{S}}$	Server's public key.

A. The Configuration Phase

In this phase, the server S generates the system parameters (q_b, a, b, q_a, P_g) required for the construction of the elliptic curve $\mathcal{C}$, shares them with other entities, and stores them in its database. The server then assigns to each smart object D_k, a unique identifier ID_{D_k}, and computes its hashed form $hID_{D_k} = \mathsf{H}_c(ID_{D_k})$, where hID_{D_k} serves as the key in the hash table used to store the identifiers of all smart objects. This process is illustrated in Fig. 2.

Server $\mathcal{S}$	**IoT Device** D_k
$(q_b, a, b, q_a, P_g) \xleftarrow{\$}$	
$ID_{D_k} \xleftarrow{\$}$	
$hID_{D_k} = \mathsf{H}(ID_{D_k})$	
$\xrightarrow{(q_b, a, b, q_a, P_g),\ ID_{D_k},\ hID_{D_k}}$	

Fig. 2. Configuration phase.

B. The Authentication Phase

This step is illustrated in Fig. 3 and proceeds as follows:

- The server $\mathcal{S}$ randomly selects $\rho_{\mathcal{S}} \in \mathbb{F}^*_{q_a}$ and computes $R_{\mathcal{S}} = \rho_{\mathcal{S}} \cdot P_g$. It then transmits $R_{\mathcal{S}}$ to the device D_k.
- Upon reception, D_k picks a random $\rho_k \in \mathbb{F}^*_{q_a}$ and calculates $R_k = \rho_k \cdot P_g$. To conceal hID_{D_k}, it derives a shared key $K = \rho_k \cdot R_{\mathcal{S}}$ and masks the identity hash as $M = hID_{D_k} \oplus K$. It then generates an authenticator $\mathsf{Auth}_k = \mathsf{H}(ID_{D_k} \parallel 2 \parallel K)$. The device sends the tuple $(R_k, \mathsf{Auth}_k, M)$ to $\mathcal{S}$.
- On receiving the message, $\mathcal{S}$ computes $K = \rho_{\mathcal{S}} \cdot R_k$ and retrieves $hID'_{D_k} = M \oplus K$. It searches the

database for the identity ID'_{D_k} corresponding to hID'_{D_k}. If no match is found, $\mathcal{S}$ ends the session after a predefined timeout to limit information leakage. Otherwise, it computes $\mathsf{Auth}'_k = \mathsf{H}(ID'_{D_k} \parallel 2 \parallel K)$ and compares it to the received Auth_k. If they differ, the session is aborted; if equal, D_k is authenticated. The server then generates $\mathsf{Auth}_{\mathcal{S}} = \mathsf{H}(ID_{D_k} \parallel 1 \parallel K)$ and sends it to D_k.

- Upon receiving $\mathsf{Auth}_{\mathcal{S}}$, the device computes $\mathsf{Auth}'_{\mathcal{S}} = \mathsf{H}(ID_{D_k} \parallel 1 \parallel K)$ and checks equality. If they match, the server is authenticated, achieving mutual authentication; otherwise, the session is terminated.

At the end of this phase, D_K and S share the session key, K, which has been computed to ensure integrity, privacy, and confidentiality during subsequent exchanges.

VI. Potential Security Weaknesses in the Fatma et al. Protocol

In the previous scheme, an adversary is able to recover the communication key K, as well as the identifier ID_{D_K} of a smart object, enabling them to intercept exchanges between the device D_K and the server $\mathcal{S}$. The attacker first obtains the hashed identifier H_{D_K} by following the steps illustrated in Fig. 4:

- The adversary $\mathcal{A}$ selects a random value $\rho_{\mathcal{A}}$ and computes $R_{\mathcal{A}} = \rho_{\mathcal{A}} \cdot P_g$. It then sends $R_{\mathcal{A}}$ to the device D_k.
- Upon receiving $R_{\mathcal{A}}$, D_k generates a random $\rho_k \in \mathbb{F}^*_{q_a}$ and computes $R_k = \rho_k \cdot P_g$. To protect hID_{D_k}, the device calculates $K = \rho_k \cdot R_{\mathcal{A}}$ and masks it as $M = hID_{D_k} \oplus K$. It also generates $\mathsf{Auth}_k = \mathsf{H}(ID_{D_k} \parallel 2 \parallel K)$, and sends $(R_k, \mathsf{Auth}_k, M)$ to $\mathcal{A}$.
- The adversary then computes $K = \rho_{\mathcal{A}} \cdot R_k$ and retrieves hID_{D_k} by computing $hID_{D_k} = K \oplus M$.

After the recovery of hID_{D_k}, the attacker can perform the following two attacks:

A. Eavesdropping Attack

Once the adversary has obtained hID_{D_k}, it stores this value in its own database. During a legitimate authentication process between $\mathcal{S}$ and D_k, the adversary can perform an eavesdropping attack to derive the session key K, as outlined in Fig. 5:

- The server $\mathcal{S}$ randomly selects a number $\rho_{\mathcal{S}}$ and computes $R_{\mathcal{S}} = \rho_{\mathcal{S}} \cdot P_g$. It then transmits $R_{\mathcal{S}}$ to D_k.
- The IoT device executes its computations and returns the tuple $(R_k, \mathsf{Auth}_k, M)$ to $\mathcal{S}$.
- The adversary intercepts this message. With knowledge of hID_{D_k} and access to $(R_k, \mathsf{Auth}_k, M)$, it can compute the communication key as $K = hID_{D_k} \oplus M$.

B. Preimage Attack

Once hID_{D_k} has been recovered, note that $hID_{D_k} = \mathsf{H}(ID_{D_k})$. Since ID_{D_k} is a scalar, a high-performance computing system could, in principle, generate candidate scalars x_k and compute $\mathsf{H}(x_k)$ until finding some x_n such that $\mathsf{H}(x_n) = \mathsf{H}(ID_{D_k})$. In that case, x_n reveals ID_{D_k}.

If ID_{D_k} is chosen with a sufficiently large bit length, such a brute-force search becomes computationally infeasible. However, this mitigation is constrained in IoT settings, where limited device storage makes it impractical to significantly increase the

VII. The Proposed ECC based Authentication Protocol

We propose an enhanced version of the protocol introduced by Fatma et al. [6] in order to address the vulnerabilities identified in Section VI. The improved scheme also supports mutual authentication between servers and multiple smart objects, provided that the servers store the identities of these objects.

The proposed protocol operates in two main phases: a configuration phase and an authentication phase.

A. The Configuration Phase

The server first creates the system parameters (q_b, a, b, q_a, P_g) and shares them with the other entities to construct the elliptic curve $\mathcal{E}$. Then, the server $\mathcal{S}$ chooses a random value $\mathsf{SK}_{\mathcal{S}} \in \mathbb{F}^*_{q_a}$ as its private key and computes its public key $\mathsf{PK}_{\mathcal{S}} = \mathsf{SK}_{\mathcal{S}} \cdot P_g$. Afterwards, the server generates and sends to each IoT device an identifier ID_{D_k} and $hID_{D_k} = \mathsf{H}(ID_{D_k})$. It also transmits its public key $\mathsf{PK}_{\mathcal{S}}$. Once this phase is completed, the server records the system parameters along with the device identifiers in its database, to be used later during the authentication phase (see Fig. 6).

B. Authentication Phase

This phase consists of three message exchanges, illustrated in Fig. 7.

- Challenge from server to device: The server $\mathcal{S}$ initiates the process by sending an authentication request (challenge) to the IoT device D_k. Upon receiving the challenge, D_k randomly selects $\rho_k \xleftarrow{\$} \mathbb{F}^*_{q_a}$, then computes $R_k = \rho_k \cdot P_g$ and $K = \rho_k \cdot \mathsf{PK}_{\mathcal{S}}$. It masks K as $M = hID_{D_k} \oplus K$ and generates the authentication tag $\mathsf{Auth}_k = \mathsf{H}(ID_{D_k} \parallel 2 \parallel K)$. Finally, it sends $(R_k, \mathsf{Auth}_k, M)$ to $\mathcal{S}$.
- Device authentication by server: The server computes $K = \mathsf{SK}_{\mathcal{S}} \cdot R_k$ and derives $hID'_{D_k} = M \oplus K$. It then searches its database for a matching ID'_{D_k} using hID'_{D_k}. If no match is found, the session is terminated after a delay to limit information leakage. If a match exists, the server calculates $\mathsf{Auth}'_k = \mathsf{H}(ID'_{D_k} \parallel 2 \parallel K)$ and compares it with Auth_k. If the values match, D_k is authenticated; otherwise, the session is aborted. Upon successful authentication, the server generates $\mathsf{Auth}_{\mathcal{S}} = \mathsf{H}(ID_{D_k} \parallel 1 \parallel K)$ and sends it to the device.

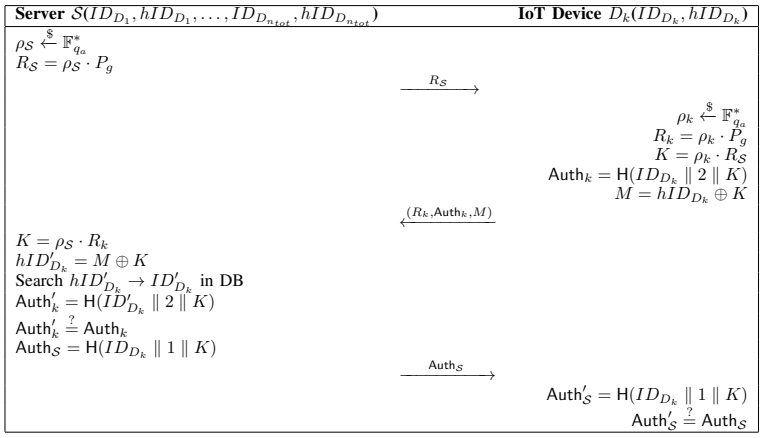


Fig. 3. Authentication phase.

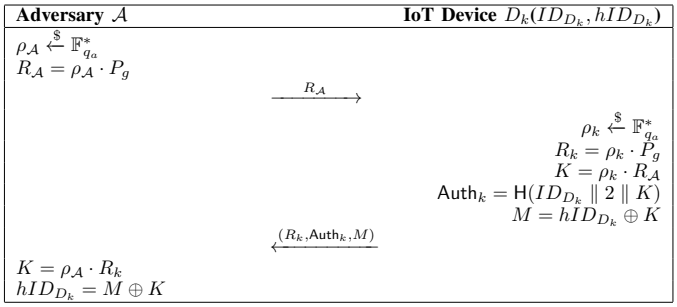


Fig. 4. Recovery of hID_{D_k}.

- Server authentication by device: After receiving $\mathsf{Auth}_{\mathcal{S}}$, the device computes $\mathsf{Auth}'_{\mathcal{S}} = \mathsf{H}(ID_{D_k} \| 1 \| K)$ and verifies it against the received value. If they match, $\mathcal{S}$ is authenticated, and mutual authentication is established; otherwise, the session is terminated.

At the conclusion of this phase, both D_k and $\mathcal{S}$ share the session key K, computed as:

$$K = \rho_k \cdot \mathsf{PK}_{\mathcal{S}} = \mathsf{SK}_{\mathcal{S}} \cdot R_k = \mathsf{SK}_{\mathcal{S}} \cdot \rho_k \cdot P_g.$$

This shared key is subsequently used to ensure integrity, confidentiality, and privacy in all subsequent communications.

VIII. Security Analysis

A. Informal Analysis

1) Mutual authentication:

a) Device authentication: An adversary is unable to generate a valid message $(R_k, \mathsf{Auth}_k, M)$ on behalf of device D_k, as they lack the necessary values hID_{D_k}, ID_{D_k}, and ρ_k. These parameters are essential for computing Auth_k and M, and can only be derived by the legitimate server $\mathcal{S}$ and device D_k.

b) Server authentication: The adversary cannot forge the legitimate server message $\mathsf{Auth}_{\mathcal{S}}$, since it requires knowledge of the identifier ID_{D_k}, which is not disclosed.

TABLE II. Results of the Verification by AVISPA

back-end	Result
CL-AtSe	SAFE
OFMC	SAFE
SATMC	INCONCLUSIVE
TA4SP	INCONCLUSIVE

2) Integrity and confidentiality: Upon completion of the protocol, the server $\mathcal{S}$ and device D_k establish a shared session key K. This key is jointly derived from $\mathsf{SK}_{\mathcal{S}}$ (generated by $\mathcal{S}$) and ρ_k (generated by D_k) for each session, thereby ensuring both message integrity and confidentiality.

3) Availability:

a) Resistance to Man-in-the-Middle (MitM) attacks: The adversary cannot obtain ID_{D_k}, hID_{D_k}, or the session key K. In contrast, the protocol of Fatma et al. [6] is susceptible to MitM attacks, as the session key K can be recovered by an adversary.

B. Formal Analysis

In this section, we conduct a formal security assessment of the proposed scheme using two widely recognized automated verification tools: AVISPA [31] and ProVerif [32].

1) HLPSL Code of the proposed protocol: The HLPSL specification of our protocol is provided in Fig. 8 and Fig. 9. For a more comprehensive explanation, including extended formal verification procedures and additional security analysis, readers are referred to our related conference publication [33].

2) AVISPA verification results: Table II summarizes the outcomes obtained after executing our HLPSL-coded protocol in the AVISPA environment. The results indicate that both the CL-AtSe and OFMC back-ends report the status SAFE, confirming the absence of detected attacks. For the SATMC and TA4SP back-ends, the outcome is INCONCLUSIVE due to unsupported operations, which implies that these modules are unable to process certain steps of the protocol rather than indicating any security weakness.

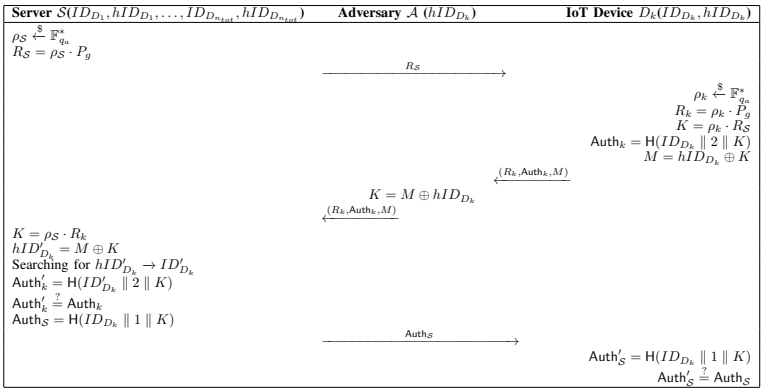


Fig. 5. Recovery of the key K.

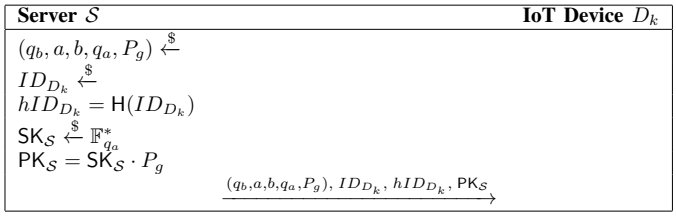


Fig. 6. ECC-based configuration phase.

C. Specification of Our Protocol using the Pi-Calculus Script in ProVerif

In this section, we will present the code of our protocol using Pi-Calculus. Fig. 10 and Fig. 11 presents the server role and smart object role.

1) The results of the verification by Proverif: Fig. 12 represents the results of our contributions using the ProVerif tool. It shows that mutual authentication is achieved, and the confidentiality of variables idt, hidt, ri, and rs is preserved.

IX. Performance Evaluation

This section presents an evaluation of the proposed scheme with respect to communication overhead, and computational effort. For the analysis, we consider a deployment consisting of n_{tot} IoT devices. We assume that the size of a scalar value is λ bits, which implies that the size of a point on the selected elliptic curve $\mathcal{E}$ is 2λ bits. Accordingly, the size of the system parameters (q_b, a, b, q_a, P_g) is taken to be 6λ bits in total. Furthermore, both the device identifier ID_{D_k} and its corresponding hashed representation hID_{D_k} are assumed to be scalar values, each occupying λ bits.

T_{PM} and T_{PA} represent the execution time required to perform elliptic curve point multiplication and point addition operations, respectively.

A. Computation Cost

In this section, we assess and compare the computational effort required by both the server (see Table IV) and the smart objects (see Table III) during the authentication process. The performance of our proposed protocol is evaluated against several existing schemes, including those presented by Liao et al. [29], Dinarvand et al. [28], Jin et al. [34], Alamr et al. [26], Zhao [30], and Fatma et al. [6].

TABLE III. Evaluation of Computation Costs for Smart Devices

approach	Computational Overheads
Zhao et al. [30]	$3 * T_{\text{PA}} + 5 * T_{\text{PM}}$
Dinarvand et al. [28]	$2 * T_{\text{PA}} + 3 * T_{\text{PM}}$
Jin et al. [34]	$1 * T_{\text{PA}} + 4 * T_{\text{PM}}$
Alamr et al. [26]	$1 * T_{\text{PA}} + 4 * T_{\text{PM}}$
Liao et al. [29]	$3 * T_{\text{PA}} + 5 * T_{\text{PM}}$
Fatma et al. [6]	$2 * T_{\text{PM}}$
Our approach	$2 * T_{\text{PM}}$

TABLE IV. Evaluation of Computation Costs for Server

approach	Computational Overheads
Zhao et al. [30]	$3 * T_{\text{PA}} + 5 * T_{\text{PM}}$
Dinarvand et al. [28]	$2 * T_{\text{PA}} + 3 * T_{\text{PM}}$
Jin et al. [34]	$2 * T_{\text{PM}}$
Alamr et al. [26]	$1 * T_{\text{PA}} + 5 * T_{\text{PM}}$
Liao et al. [29]	$3 * T_{\text{PA}} + 5 * T_{\text{PM}}$
Fatma et al. [6]	$2 * T_{\text{PM}}$
Our approach	$1 * T_{\text{PM}}$

As shown in Table IV, our proposed improvement achieves lower server computation costs than the compared protocols, while maintaining a database search complexity of $O(1)$, similar to the approach of Fatma et al. [6].

B. Transmission Overhead

The transmission Overhead is assessed by determining the bit length of all messages exchanged during the authentication phase, along with counting the total number of message transmissions.

In Table V, our enhancement is the best protocol in terms of computation cost at the server level.

X. Practical Analysis

To evaluate the feasibility and performance of the proposed ECC-based mutual authentication protocol, a real-world imple-

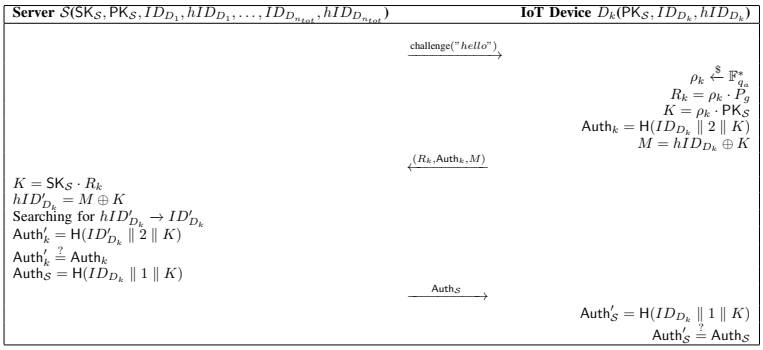


Fig. 7. ECC-based authentication phase.

Server role

```
role serverRole(A,B : agent, Q,Idx,HashX,Txt : text, Msgs : message, F,HashFunc : hash_func, SEND,RECEIVE : channel(dy))
played_byA
def =
    local
        Status : nat, Rnd,Key : text,
        Rq,Msg,AuthA,AuthB : message
        const sec_rnd,sec_idx,sec_hashx,auth_b,auth_a : protocol_id
    init
        Status := 0
    transition
        1.Status = 0 ∧ RECEIVE(start) ⇒ Status' := 2 ∧ Rnd' := new()
                                       ∧SEND(Rnd)
        2.Status = 2 ∧ RECEIVE(Rq', AuthB', Msg')
                     ∧Key' = HashFunc(Txt, Rq')
                     ∧HashX = xor(Msg', Key')
                     ∧AuthB = F(Idx'.2.Key') ⇒ Status' := 4 ∧ AuthA' := F(Idx.1.Key')
                                                ∧SEND(AuthA')
                                                ∧secret(Idx, sec_idx, A)
                                                ∧secret(HashX, sec_hashx, A)
                                                ∧request(A,B, auth_b, AuthB')
                                                ∧witness(A,B, auth_a, AuthA')
end role
```

Fig. 8. Server role of HLPSL.

Object role

```
role objectRole(B,A : agent, Q,Ident,HashIdent : text, MsgList : message, F,HashFunc : hash_func, SEND,RECEIVE : channel(dy))
played_byB
def =
    local
        Status : nat, Random,Key : text,
        ReqIdent,RespIdent,Msg,AuthA,AuthB : message
        const sec_random,sec_ident,sec_hashident,auth_a,auth_b : protocol_id
    init
        Status := 1
    transition
        1.Status = 1 ∧ RECEIVE(RespIdent') ⇒ Status' := 3 ∧ Random' := new()
                                           ∧ReqIdent' := HashFunc(Random', Q)
                                           ∧Key' := HashFunc(Random', MsgList)
                                           ∧AuthB' := F(Ident.2.Key')
                                           ∧Msg' := xor(HashIdent, Key')
                                           ∧SEND(ReqIdent', AuthB', Msg')
                                           ∧secret(Ident, sec_ident, B)
                                           ∧secret(HashIdent, sec_hashident, B)
                                           ∧secret(Random', sec_random, B)
                                           ∧witness(B,A, auth_a, AuthB')
        2.Status = 3 ∧ RECEIVE(AuthA')
                     ∧AuthA' = F(Ident.1.Key) ⇒ Status' := 5 ∧request(B,A, auth_b, AuthA')
end role
```

Fig. 9. Role of object B.

mentation was conducted using two heterogeneous physical devices: a Raspberry Pi 3 Model B+ representing the IoT object, and a standard personal computer equipped with an Intel Core i5 (5th generation) processor functioning as the central server. The Raspberry Pi, operating under a Linux-based Raspbian OS, was chosen to simulate the constraints typical of embedded IoT devices in terms of memory, CPU capacity, and energy consumption. The server ran Ubuntu 20.04 and was responsible for handling the more computationally intensive cryptographic verifications and key exchange processes. The protocol logic was fully developed in Python 3 using the ecdsa cryptographic library, which provides implementations of elliptic curve operations and ECDSA signatures. Experiments were conducted using a range of elliptic curves to evaluate their impact on the protocol's performance. The tested curves include: NIST192p, NIST224p, NIST256p, SECP256k1, BRAINPOOLP160r1, BRAINPOOLP192r1, BRAINPOOLP256r1, and Ed25519. These curves were used to measure the execution time of the authentication phase for each configuration, and the results obtained are presented in Fig. 13.

Server Role

```
(* Role of the Server S *)
let SERVERS =
    new rs : bitstring;
    let RS = mecc(rs, p) in
    out(ch, RS);
      in(ch, (Ri : bitstring, authi : bitstring, hidt_recv : bitstring));
    let K = mecc(Ri, rs) in
    if authi = h(concat(idt, concat(v2, K))) then
        let AUTHS = h(concat(idt, concat(v1, K))) in
        event beginS (AUTHS);
        out(ch, AUTHS);
        event endS (AUTHS).
```

Fig. 10. Server role.

Smart Object Role

```
(* Role of the Smart Object I *)
let SOI =
    in (ch, RS : bitstring);
    new ri : bitstring;
    let RI = mecc(ri, p) in
    let K = mecc(ri, RS) in
    let AUTHI = h(concat(idt, concat(v2, K))) in
    event beginT (AUTHI);
    out (ch, (RI, AUTHI, hidt));
    in (ch, auths : bitstring);
    if auths = h(concat(idt, concat(v1, K))) then
        event endT (AUTHI).
```

Fig. 11. Smart object role.

```
------------------------------------------------------------
Verification summary:

Query not attacker(idt[]) is true.

Query not attacker(ri[RS_1 = v,!1 = v_1]) is true.

Query not attacker(rs[!1 = v]) is true.

Query inj-event(beginS(x)) ==> inj-event(beginT(x)) is true.

Query inj-event(endT(x)) ==> inj-event(endS(x)) is true.

Query inj-event(endS(y)) ==> inj-event(beginS(x)) && inj-event(beginT(x)) is true.

------------------------------------------------------------
```

Fig. 12. Results of the verification by Proverif.

TABLE V. COMPARISON OF TRANSMISSION OVERHEAD

approach	S → O	O → S	Total (bits)	Number of Messages
Alamr et al. [26]	$4 * \lambda$	$6 * \lambda$	$10 * \lambda$	3
Dinarvand et al. [28]	$5 * \lambda$	$4 * \lambda$	$9 * \lambda$	4
Jin et al. [34]	$4 * \lambda$	$3 * \lambda$	$7 * \lambda$	3
Zhao et al. [30]	$4 * \lambda$	$4 * \lambda$	$8 * \lambda$	3
Liao et al. [29]	$4 * \lambda$	$4 * \lambda$	$8 * \lambda$	3
Fatma et al. [6]	$4 * \lambda$	$3 * \lambda$	$7 * \lambda$	3
Our proposed	$4 * \lambda$	$2 * \lambda$	$6 * \lambda$	3

This implementation setup allowed for real-time testing of the protocol under realistic hardware constraints and provided concrete insights into its computational efficiency and practicality for IoT deployments.

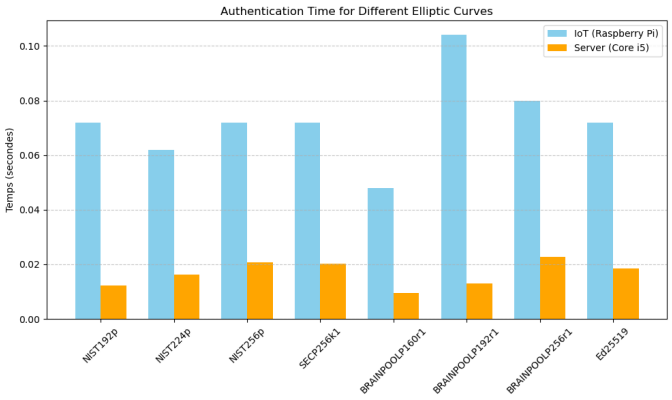


Fig. 13. Authentication time for different elliptic curves.

A. Computational Cost of Our Protocol

The average execution time of our authentication protocol, measured across the full exchange phase between the IoT device (Raspberry Pi) and the server (Core i5), is approximately 0.089348 seconds, consisting of 0.072703 seconds on the IoT side and 0.016645 seconds on the server side. These values were obtained by computing the mean execution time across a range of standardized elliptic curves.

This implementation demonstrated that our protocol successfully reduced the execution time compared to the protocol proposed by Fatma et al., particularly during the authentication phase, as illustrated in Fig. 14 . This noticeable reduction in processing time highlights the effectiveness of our approach, which combines computational efficiency with robust security. As a result, the proposed protocol is well suited for real-time execution on resource-constrained IoT platforms, while ensuring secure and mutual authentication.

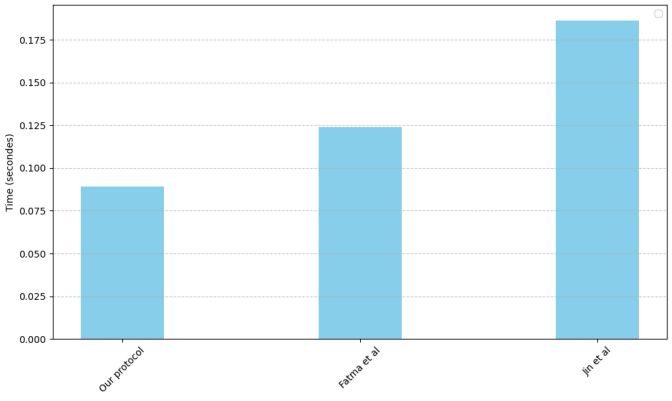


Fig. 14. Comparison of computation costs.

XI. CONCLUSION

This research presents a comprehensive study of an efficient protocol originally proposed by Fatma et al. Our security

analysis revealed several critical vulnerabilities that could compromise the robustness of the protocol in real-world IoT deployments. In response to these weaknesses, we proposed a reinforced version of the protocol, focusing on both improving security guarantees and reducing computational and communication overhead. To validate the proposed enhancements, we not only employed formal verification tools, such as AVISPA and ProVerif but also carried out a practical implementation using heterogeneous devices—specifically, a Raspberry Pi 3 Model B+ and a standard PC—to simulate constrained IoT environments. This real-world experimentation allowed us to analyze execution times and resource consumption across a variety of elliptic curves. The results, drawn from both simulation and implementation, clearly demonstrate that the enhanced protocol significantly improves upon the original in terms of resistance to known attacks and performance efficiency. These outcomes confirm the protocol's relevance and feasibility for secure authentication in IoT systems and highlight the value of combining formal analysis with empirical validation to design resilient and practical security solutions.

REFERENCES

[1] H. Zougagh, N. Idboufker, and Y. Saadi, "Trust-based intrusion detection for multi-path olsr protocol," in *Proceedings of the 12th International Conference on Soft Computing and Pattern Recognition (SoCPaR 2020)*, vol. 12. Springer International Publishing, 2021, pp. 690–705.

[2] H. Zougagh, N. Idboufker, Y. El Mourabit, Y. Saadi, and S. Elouaham, "Avoiding wormhole attack in manet using an extending network knowledge," in *Innovations in Bio-Inspired Computing and Applications: Proceedings of the 11th International Conference on Innovations in Bio-Inspired Computing and Applications (IBICA 2020) held during December 16-18, 2020*, vol. 11. Springer International Publishing, 2021, pp. 217–230.

[3] Z. Elhadari, H. Zougagh, N. Idboufker, and M. Ech-chebaby, "Survey on the adoption of blockchain technology in internet of things environments: Techniques, challenges and future research directions," *IAENG International Journal of Computer Science*, vol. 52, no. 1, pp. 59–89, 2025.

[4] M. Naji and H. Zougagh, "Deep learning models for cybersecurity in iot networks," in *International Conference on Business Intelligence*. Springer Nature Switzerland, July 2023, pp. 29–43.

[5] H. Zougagh, N. Idboufker, R. Zoubairi, and R. El Ayachi, "Prevention of black hole attacks on mobile ad hoc networks through intrusion detection systems," *International Journal of Business Data Communications and Networking (IJBDCN)*, vol. 15, no. 2, pp. 73–91, 2019.

[6] F. Merabet, A. Cherif, M. Belkadi, O. Blazy, E. Conchon, and D. Sauveron, "New efficient m2c and m2m mutual authentication protocols for IoT-based healthcare applications," vol. 13, no. 2, pp. 439–474, 03 2020. [Online]. Available: https://doi.org/10.1007/s12083-019-00782-8

[7] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, vol. 48, pp. 203–209, 1987.

[8] S. Pote, V. Sule, and B. K. Lande, "Arithmetic of koblitz curve *Secp256k1* used in bitcoin cryptocurrency based on one variable polynomial division," 04 2019. [Online]. Available: https://papers.ssrn.com/abstract=3367674

[9] J. H. Silverman and J. Suzuki, "Elliptic curve discrete logarithms and the index calculus," in *Advances in Cryptology — ASIACRYPT'98*, K. Ohta and D. Pei, Eds. Springer Berlin Heidelberg, 1998, pp. 110–125.

[10] P. Gaudry, "Index calculus for abelian varieties of small dimension and the elliptic curve discrete logarithm problem," vol. 44, no. 12, pp. 1690–1702, 12 2009. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S074771710800182X

[11] A. Srivastava and A. Kumar, "A review on authentication protocol and ecc in iot," in *2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, 2021, pp. 312–319.

[12] B. Preneel, "Cryptographic hash functions," vol. 5, no. 4, pp. 431–448, 1994, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/ett.4460050406. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/ett.4460050406

[13] X. Wang and H. Yu, "How to break MD5 and other hash functions," in *Advances in Cryptology – EUROCRYPT 2005*, R. Cramer, Ed. Springer Berlin Heidelberg, 2005, pp. 19–35.

[14] N. Kheshaifaty and A. Gutub, "Preventing multiple accessing attacks via efficient integration of captcha crypto hash functions," vol. 20, pp. 16–28, 09 2020.

[15] Rohit, S. Kamra, M. Sharma, and A. Leekha, "Secure hashing algorithms and their comparison," in *2019 6th International Conference on Computing for Sustainable Global Development (INDIACom)*, 2019, pp. 788–792.

[16] A. A. Yaseen, K. Patel, A. A. Yassin, A. J. Aldarwish, and H. A. Hussein, "Secure electronic healthcare record using robust authentication scheme," *IAENG International Journal of Computer Science*, vol. 50, no. 2, pp. 468–476, 2023.

[17] S. Boonkrong, "Security analysis and improvement of a multi-factor biometric-based remote authentication scheme," *IAENG International Journal of Computer Science*, vol. 46, no. 4, pp. 713–724, 2019.

[18] Y. Park, K. Park, K. Lee, H. Song, and Y. Park, "Security analysis and enhancements of an improved multi-factor biometric authentication scheme," *International Journal of Distributed Sensor Networks*, vol. 13, no. 8, p. 1550147717724308, 2017. [Online]. Available: https://doi.org/10.1177/1550147717724308

[19] L. Zhang, Y. Zhang, S. Tang, and H. Luo, "Privacy protection for e-health systems by means of dynamic authentication and three-factor key agreement," *IEEE Transactions on Industrial Electronics*, vol. 65, pp. 2795–2805, 2018.

[20] Q. Jiang, J. Ma, C. Yang, X. Ma, J. Shen, and S. A. Chaudhry, "Efficient end-to-end authentication protocol for wearable health monitoring systems," vol. 63, pp. 182–195, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045790617305128

[21] R. Amin, S. H. Islam, G. P. Biswas, M. K. Khan, and N. Kumar, "A robust and anonymous patient monitoring system using wireless medical sensor networks," vol. 80, pp. 483–495, 03 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X16301509

[22] X. Li, J. Peng, S. Kumari, F. Wu, M. Karuppiah, and K.-K. Raymond Choo, "An enhanced 1-round authentication protocol for wireless body area networks with user anonymity," vol. 61, pp. 238–249, 07 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045790617303099

[23] J. Liu, L. Zhang, and R. Sun, "1-RAAP: An efficient 1-round anonymous authentication protocol for wireless body area networks," vol. 16, no. 5, p. 728, 05 2016, number: 5 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://www.mdpi.com/1424-8220/16/5/728

[24] N. Yessad, S. Bouchelaghem, F.-S. Ouada, and M. Omar, "Secure and reliable patient body motion based authentication approach for medical body area networks," vol. 42, pp. 351–370, 12 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1574119217303048

[25] C. Jin, C. Xu, X. Zhang, and F. Li, "A secure ECC-based RFID mutual authentication protocol to enhance patient medication safety," vol. 40, no. 1, p. 12, 10 2015. [Online]. Available: https://doi.org/10.1007/s10916-015-0362-8

[26] A. A. Alamr, F. Kausar, J. Kim, and C. Seo, "A secure ECC-based RFID mutual authentication protocol for internet of things," vol. 74, no. 9, pp. 4281–4294, 09 2018, company: Springer Distributor: Springer Institution: Springer Label: Springer Number: 9 Publisher: Springer US. [Online]. Available: https://link.springer.com/article/10.1007/s11227-016-1861-1

[27] N. Mehibel and M. Hamadouche, "A new approach of elliptic curve diffie-hellman key exchange," in *2017 5th International Conference on Electrical Engineering - Boumerdes (ICEE-B)*, 2017, pp. 1–6.

[28] N. Dinarvand and H. Barati, "An efficient and secure RFID authentication protocol using elliptic curve cryptography," vol. 25, no. 1, pp. 415–428, 01 2019. [Online]. Available: https://doi.org/10.1007/s11276-017-1565-3

[29] Y.-P. Liao and C.-M. Hsiao, "A secure ECC-based RFID authentication scheme using hybrid protocols," in *Advances in Intelligent Systems and Applications - Volume 2*, ser. Smart Innovation, Systems and Technologies, J.-S. Pan, C.-N. Yang, and C.-C. Lin, Eds. Springer, 2013, pp. 1–13.

[30] Z. Zhao *et al.*, "A secure rfid authentication protocol for healthcare environments using elliptic curve cryptosystem," *Journal of Medical Systems*, vol. 38, no. 5, p. 46, 2014. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/24756871/

[31] L. Viganò, "Automated security protocol analysis with the AVISPA tool," vol. 155, pp. 61–86, 05 2006. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1571066106001897

[32] B. Blanchet, "Automatic verification of security protocols in the symbolic model: The verifier ProVerif," in *Foundations of Security Analysis and Design VII: FOSAD 2012/2013 Tutorial Lectures*, ser. Lecture Notes in Computer Science, A. Aldini, J. Lopez, and F. Martinelli, Eds. Springer International Publishing, 2014, pp. 54–87. [Online]. Available: https://doi.org/10.1007/978-3-319-10082-1_3

[33] M. Ech-chebaby, H. Zougagh, H. Garmani, Z. El-hadari, and N. Idboufker, "Enhanced ecc-based authentication protocol for iot devices m2c," in *Proceedings of the 4th International Conference on Advances in Communication Technology and Computer Engineering (ICACTCE'24)*, C. Iwendi, Z. Boulouard, and N. Kryvinska, Eds. Cham: Springer Nature Switzerland, 2025, pp. 154–169.

[34] C. Jin, C. Xu, X. Zhang, and F. Li, "A secure ECC-based RFID mutual authentication protocol to enhance patient medication safety," vol. 40, no. 1, p. 12, 01 2016. [Online]. Available: https://doi.org/10.1007/s10916-015-0362-8

QoS-Aware Deployment and Synchronization of Digital Twins Over Federated Cloud Platforms for Smart Infrastructure Monitoring

M V Narayana[1], Naveen Reddy N[2], Madhu S[3], Madhu T[4], Niladri Sekhar Dey[5], Sanjeev Shrivastava[6]
Professor, Guru Nanak Institutions Technical Campus, Hyderabad, Telangana, India[1, 3]
Assistant Professor, Guru Nanak Institutions Technical Campus, Hyderabad, Telangana, India[2]
Professor, Sri Venkateswara Engineering College, Suryapet, Telangana, India[4]
Department of AI&DS, B V Raju Institute of Technology, Narsapur, Telangana, India[5]
Professor, UIET, Guru Nanak University, Hyderabad, India[6]

***Abstract*—This electronic increasingly, Digital Twin (DT) systems are being leveraged in smart infrastructure settings (e.g., structural health monitoring, intelligent traffic controls, and distributed utility networks). Yet, the available solutions all face hurdles that can prevent real-time synchronization of DT instances across federated cloud platforms, primarily due to latency variation, quality of service (QoS) assurance, and stale data, which are all consequences of heterogeneous computer environments. Most solutions depend on static cloud-only models of deployment, with no option for dynamic negotiation of resources. These provide long update times (typically greater than 200ms), low accuracy rates, and low real-time responsiveness. Additionally, traditional DT models were not developed with multi-regional deployment or QoS workloads in mind. In this work, a QoS-Aware Federated Digital Twin Orchestration Framework (Q-FDTO) is designed to allow latency-critical monitoring of infrastructure across different federated cloud regions, through the integration of a hybrid edge-cloud control plane, adaptive synchronization, jitter consideration for observed intervals, and dynamic resource allocation via reinforcement learning for defined QoS Service Level Objectives (SLOs). This system was evaluated on a smart city testbed of 1200 sensor nodes. The testbed monitored sensor readings for structural strain, vibration, and traffic density across twelve locations. The digital twin pipeline is comprehensive [i.e., (i) ingestion via Wi-Fi MQTT, (ii) stream fusion of all the sensor readings via Kalman filtering, and (iii) twin modeling of prediction, through a temporal graph convolutional network (T-GCN)]. To assess performance, sync policies were evaluated on metrics for average update latency (ms), sync drift (ms), and data consistency rate (%). The results demonstrate that Q-FDTO had an average update latency of 87.3 ms, reduced from 194.6 ms, and a 96.2% consistency rate across federated nodes with less than 2.5% sync drift over 10-minute intervals, showing Q-FDTO architecture ability for network boundaries and also compatible with AWS Outposts and Azure Arc hybrid cloud environments. It establishes a scalable and practical approach to latency-sensitive DT deployments in the realm of smart infrastructure systems.**

Keywords*—QoS-aware digital twins; federated cloud synchronization; smart infrastructure monitoring; latency-constrained orchestration; edge-assisted deployment

I. Introduction

The worldwide expansion of smart infrastructure, including urban mobility, structural health monitoring, energy-resilient buildings, and smart water systems, has led to rapid growth in sensor-rich contexts. Smart infrastructure comprises a system of sensor-actuator networks that, together, stream telemetry data to incorporate into real-time decisions, fault identification or preventative maintenance. Like digital twins, a form of monitoring and decision-making technology, as of 2023, over 65% of cities worldwide have implemented some form of monitoring of their infrastructure, typically in partnership with private companies focused on the technology market and cities taking on digital, scaled transformation initiatives [1]. Digital twins are models of a real infrastructure component, built with real-time data, configuration specifics, and geographic data to create a virtual version of the real component. Digital twins serve as predictive devices or operational dashboards that facilitate identifying and proactively responding to faults, load distributions, and capacity optimization. For instance, in smart traffic networks, digital twins may be used to model how vehicles flow through junctions and when it would be the most beneficial moment to change the program of a stoplight to relieve congestion in relation to other stoplights. Alternatively, in a structural health assessment context, data streams of strain and vibration measures expected in real-time could be reference frames for models predicting structural fatigue and failures. Moreover, despite the developing importance of the topic, limitations concerning effectiveness of such systems to perform at the willingness scale, while managing data processing and modeling latencies, and the shape and distribution of hardware and network topology, along with time-varying flows, and other data properties continue to challenge; such as any fidelity is time varying, and the highly dynamic conditions experienced while monitoring behavior in physical systems, like a vehicular flow, or structural loads affected by weather, ends up triggering (at least in pilot implementations) unattended faults and periods of uncertainty; further, as much as these restrictions continue to evolve, there are severe constraints regarding data transfer limits and waiting system processing. The monitored physical system itself is unable to be optimized for low-latency times or any other characteristics, and intervention will be needed in cases of

disparity across systems when using any non-latency-constrained twin. However, any twin frame subjectively based on low-latency and high-consistency telemetric transfer parameters would still be deployed across architectures of many different thermal characteristics and sensing capabilities that would not be able to transfer motion behavior parameters without reduced fidelity, delays in telemetric transfer, or potentially false triggering with an operator decision. In addition to these factors, the emergence of federated cloud environments—where multiple cloud regions or providers are harmonized to sustain distributed applications—is motivating more resilient, cost-efficient, and geo-political-compliant smart infrastructure operators to abandon single-vendor cloud deployments. This generates another layer of complexity, which is synchronizing digital twin states across federated infrastructures with strict QoS constraints on both latency (sub-100ms), jitter (variance under 20ms), and high-frequency data update rates (e.g., 10–50Hz in traffic systems).

Digital twins, in the context of cyber-physical systems (CPS), identify real-time (or near real-time), dynamic digital models that are continuously updated to reflect changes in their real-world counterpart, based upon incoming telemetry data. Digital twins are not mere static replicas; they change, evolve, and forecast based on environmental interactions. Additionally, they can be predictive, prescriptive, or real-time descriptive in nature. A robust digital twin architecture includes a data ingestion layer, a processing and simulation layer, and a visualization/control interface. Furthermore, cloud computing provides a natural backbone for managing and hosting digital twin workloads. Even at the level of public cloud platforms, such as AWS IoT Twin Maker, Azure Digital Twins, and Siemens Mind Sphere, there are off-the-shelf APIs to support digital twin management in real-time. However, as infrastructure deployments become more latency-sensitive, the execution of some operations needs to be offloaded to edge computing layers, such as complete processing offloading to gateways or fog nodes. While edge layers function as intermediaries, they enable pre-processing, caching, and low-latency decision enforcement closer to the physical assets. Within the previous context, federated cloud platforms represent a paradigm where multiple cloud services—across vendors or regions—are abstracted and managed as a singular logical fabric. These frameworks are necessary for multi-tenant smart city deployments or cross-jurisdictional monitoring applications that are sensitive to data sovereignty, failover support, or local compliance. The syncing of digital twin models across federated counterparts creates new challenges of data staleness, model drift, and lag in coordination. Studies on syncing protocols (delta propagation, distributed consensus - RAFT or Paxos protocols, or event-based replication) exist for traditional distributed systems, but they are not appropriate for latency-constrained, high-frequency updates mandated by modern smart infrastructure. Moreover, there remain unexplored layers of opportunity for optimizing adaptive sync intervals based on workload dynamics or input data stream entropy.

Even with the existence of digital twin frameworks and cloud orchestration infrastructure, existing systems address a number of key limitations to achieve real-time quality-of-service (QoS) as required by monitoring of smart infrastructure. First, static cloud-deployed digital twin infrastructures have deterministic latency floors - specifically when a cloud region is distant from the asset(s) being monitored - and observed latency of greater than 150-200ms by measurement in standard HTTP or an MQTT broker pipeline across multi-region cloud environments. With existing latency more than 30 to 50ms for straining monitoring bridges where stress is varied prior to a prediction of critical failure, the dynamic must be judged in real-time if the twin can cease predictive control. Second, the cyclical determination of sync intervals represents a choke point that has not been addressed in any existing framework. Existing frameworks simply synchronize digital twin instances at fixed or other uniform intervals, while ignoring input data dynamics, workload entropy, or operational criticality. For instance, a vibration sensor operating in a static load condition will not necessitate the same update frequency as a vibration sensor that is detecting load spikes during busy hours. Not recognizing the differences between these two scenarios could result in either under-sampling important events or over-burdening the network with updates that provide limited value. Third, while eventual consistency or periodic snapshotting can usually maintain cross-region consistency in federated cloud contexts, the methods are seldom appropriately scaled or timely enough for DT use cases that require consistent views across multiple visualizations or control surfaces (e.g., when emergency response teams in different locations in the field depend on the same digital twin state). Current cloud-native DT tools do not effectively handle replication lag or update collisions. Fourth, there is often a lack of QoS-awareness with cloud orchestration. Cloud orchestrators will focus on cane, storage, or reducing costs, but do not factor in QoS requirements at the application level, like triggering updates within a latency threshold, tolerating jitter, or managing queues for updates based on a priority system, causing resource under-provisioning and/or incorrect scaling policies during critical operational periods. Lastly, learning-based synchronization models are seldom integrated. Current systems use static thresholds or heuristic rules to manage the rate of twin updates; however, reinforcement learning (RL) based controllers would optimize synchronization strategies by learning workload patterns, network conditions, and scores around criticality. Their absence in production-grade DT orchestration is an under-utilization of an opportunity for intelligent automation.

In this study, we will introduce a QoS-Aware Federated Digital Twin Orchestration Framework (Q-FDTO), developed to facilitate scalable, latency-constrained, and consistency-assured digital twin deployments in federated cloud environments. The Q-FDTO introduces a multi-layer architecture that consists of: 1) an edge-assisted ingestion and pre-processing module, 2) a cloud-central twin modeling pipeline utilizing T-GCNs to develop time-series predictions, and 3) a reinforcement-learning based adaptive synchronization controller. The framework runs on real-time telemetry from multiple Intelligent Infrastructure testbed sites made up of structural strain gauges, accelerometers, and vehicular traffic sensors. MQTT is used for edge-to-cloud transport, with inter-cloud twin synchronization managed by a lightweight, custom delta-protocol approach that uses QoS

driven control signals. The RL-based controller manages adjustment installation level and priority queue based on entropy, and latency feedback, along with criticality classification. Q-FDTO is designed to operate in conjunction with existing federated cloud platforms like Azure Arc or AWS Outposts already in use or planned for use, enabling an edge-cloud hybrid deployment for multi-regionality orchestration. The synchronization layer aims to optimize for less than 100ms update latency, with > 95% data consistency and < 2.5% average drift over 10-minute evaluation windows. The entire stack is deployable as containerized microservices using Kubernetes and supports horizontal scaling under workloads.

II. Related Works

Digital twins (DTs) facilitate real-time mirroring and simulation, and control of physical assets and stakeholder interactions, and are becoming a ubiquitous capability of modern cyber-physical systems. In areas such as transportation, smart manufacturing, and energy management, DTs enable predictive insights and facilitate in-situ operational decisions generated from innumerable timely data. Despite their burgeoning deployment and ET use cases, substantial unsolved challenges exist regarding the viability of digital twins in federated cloud frameworks (i.e., heterogeneous compute and storage across cloud vendors and physical locations), imposed by latency requirements, data synchronizations over achievable bandwidths, and consistency across instances of federated DTs. Similar challenges arise while introducing the use of edge computing, artificial intelligence (AI) model generation, and time-sensitive networking (TSN) in conjunction with DTs, as conventional synchronization and orchestration methods will not scale with beginning DTs and the user experiences designed into them. In advancing the state of practice in federated digital twin systems, it is important to examine recent works addressing task offloading, cloud-edge integration, point cloud reconstruction, and privacy-aware digital twin orchestration. X. Tan et al. [1] tackled determining task scheduling in digital twin-enabled vehicular networks, proposing a framework and an adaptive scheduling algorithm within a cloud-native vehicular communication framework. Their adaptive scheduling algorithm, built upon DTs of vehicles and traffic status, allows forecasting computational load and dynamically scheduling tasks or transferring tasks from a cloud node or agenda to the best edge node to optimize performance. The system was evaluated based on simulation models by vehicular communication topologies with relationship to task and cloud-edge demands, which resulted in lower latency and greater throughput responses along the fluctuating processing task demands. However, their proposed solution is tightly coupled to vehicular networks and has limited applicability to other infrastructure applications, such as structural health monitoring or urban sensing grids. Their solution is limited to vehicular domains and for not represent wider application use cases, particularly in cases where synchronization across multi-cloud zones needs to be enforced. Y. Gao et al. [2] proposed a LiDAR-based approach to volumetric damage evaluations and digital twin synchronization in construction environments. In their approach, they utilized a high-resolution point cloud dataset to identify deformations and structural anomalies in real time, which then could be synchronized dynamically objects contained within each architectural domain site for visualization, control, and remote monitoring. Using real-world construction datasets, they demonstrated that the framework exhibited accurate 3D map modeling with some mapping, including damage localization. However, the synchronized mechanism lacked fluidity and was limited by batching throughout a given time series, thus missing important opportunities for high-velocity, time-constrained applications requiring sub-100ms updates or dynamic synchronization windows based on sensing event entropy, etc.

J. Xie et al. [3] developed a cloud-edge collaborative dual digital twin model for electric vehicle (EV) battery systems. The dual cloud-edge system implements Lyapunov-based incremental learning to adaptively update each digital twin model for battery state-of-health associated with environments spawned by both the cloud (OEM) and edge operational (repo / transportation / logistics) conditions. Using real telemetry data from an EVs battery manufacturer, the dual cloud-edge system domain showed faster convergence and battery cell fault occurrence identified assistively with position in the digital twin model and features. While effective using the learning-based synchronization combination, the implementation was limited to battery management in one vehicle, directed at battery utilization and did not cascade out to any other federated or synchronized structural protocols managed by multiple regions across geography. The study also did not address the key elements around consistency across cloud zones identified with these domains, which are critical components when managing regional infrastructures at city levels.

Y. Li et al. [4] fused scene-based vision data with the point cloud simulator data to generate a common model to assert a digital twin for shield tunnel construction. Their method fused multiple modes of sensing to increase accuracy in detecting tunnel segment displacements and changes in geometric deformation. Real data from multiple projects for tunnel construction, inhabited in-use model improvements based on the proposed model, improving hybrid performance conditions in increasing form modeling precision. However, the study was primarily focused on offline synchronization and post-processed data application for visualization modeling at construction sites and did not provide the capabilities of real time, in- situ sensory response mechanisms aligned to railing on the intended site, or settings where dynamic synchronization would provide performance for high frequency types of models or remote adaptations for environmental type quality interaction.

Z. Hao et al. [5] focused on the pose reconstruction for hydraulic support groups using digital twins integrated with point cloud data. The reconstruction was enabled using geo-extraction mechanisms to quart this point data to object model parameters which volume measure spatial distance as part of construction environment framework - thus compose it in accordance with digital twin frame performance research engineering volume determination and measuring reconstruction outcomes in mining conditions and practices. Nonetheless, as a reconstruction posing strategy, this approach achieved a high level of accuracy with object model validation

performance when applied in field data or benchmarks. While the use of geometrical models advanced precision gap detection or modeling around consistency performance model, the PO-PO approaches representing digital twin models were not described for cross usage situations across multi-sites or federated types of multi-model synchronization examples - nonetheless, “adaptive or quality of service-based orchestration for stability in operational load variations was not captured or used". H. Yi [6] addressed the issues of privacy and security in the cloud storage of digital twin-based medical records. The study proposed a novel cloud encryption scheme that preserves the confidentiality of medical DT data while retaining interoperability among cloud devices and thereby accessibility from multiple platforms. The simulation results exhibited improved robustness to external attacks towards unauthorized data access, while storage space was used more efficiently through deduplication. However, the scheme addressed static (as opposed to dynamic) cloud architectures, leaving current digital twin deployments without real-time synchronization dynamics or latency guarantees, both are vital in real-time healthcare decision making and cloud federation deployments.

Y. Kalyani et al. [7] examined a novel digital twin deployment in the context of smart agriculture, utilizing a Cloud–Fog–Edge infrastructure model. The DT framework uses sensor-based DT instances to monitor crop health, soil moisture and surrounding environmental conditions, while distributed control was achieved through a hierarchy of offloading strategies. The proposed cloud mobile system was tested on existing IoT simulators and edge nodes that resulted in low latency and efficient data offloading. However, the proposal was built around domain-specific assumptions to agriculture and would neither scale in using federated cloud nor address the issues of (global) consistency and synchronizing across acquisitions and exchanging data across twins in a multi-region topology to address the societal dimensions of smart agriculture. C. L. Stergiou et al. [8] discussed the broader challenges associated with privacy and security of data in big data systems with an IoT-enabled paradigm and the Digital Twin context. They introduced a trust-aware data processing model within a cloud-based DT ecosystem emphasizing data access control and secure communication channels for exchanging data. The article provided a theoretical basis for the integration of trust models; however, it offered limited experimental validation and did not provide a real-time synchronization solution ideally fit for accommodating uncertainty in data rates and QoS limitations in infrastructure systems. J. Costa et al. [9] published an experimental research study of the effects of software aging on orchestration systems based on Kubernetes, within the context of urban air mobility applications of digital twin infrastructures. By deploying digital twin services within a containerized ecosystem and measuring the lifecycle behavior of this deployed system, they documented performance degradation behavior, inefficiencies of containers in the orchestration system, and evidence of memory leaks as characteristics of operation. This research is valuable for exposing operational pitfalls associated with long-life digital twin systems when deployed within a cloud-native context. That said, J. Costa et al. [9] made no practical remedies for real-time synchronization drift, nor did the paper convey a research agenda for adaptive resource scheduling for digital twins based on underlying QoS metrics inherent in federated infrastructures. E. Wang et al. [10] examined cloud-based digital twin models for scenarios where emergency healthcare is required while emphasizing latency in data storage and retrieval for post-crisis scenarios. The architecture utilized distributed database backends with front-mirrored digital twin instances and ensured data was made readily available. The authors demonstrated retrieval improvements versus a centralized data system, but the paper presents a strategy for synchronizing across cloud zones, nor did it test the performance of the architecture in high-entropy sensor conditions or operate within latency constraints inherent in emergency response workflows.

L. Gu et al. [11] developed a collaborative offloading method for digital twin-enabled edge-cloud systems on the Internet of Vehicles (IoV). The developed mechanism aimed to balance computational work between edge nodes and cloud servers and was powered by a workload dynamically partitioning process that reflects vehicular mobility and deadlines of individual task workloads. The authors evaluated the strategies via modeling and simulation and reported improved rates of task completion and average response times. As might be expected, "collaborative measurements, fault tolerance, and robust system behavior" were only tested in a vehicular digital twin use case scenario, nor did they address robustness in synchronization across multiple federated zones, nor could the strategy have generalized to stationary infrastructure contexts requiring prolonged consistency in operation. J. Dong et al. [12] developed a Mixed Cloud Control Testbed for validating vehicle-road-cloud integration with digital twins built up on and simulating real-world dynamic road conditions and vehicle behavior in a hybrid simulation environment. The authors employed the Mixed Cloud Control Testbed as a validation platform for cloud-side control logic and provided validation of the system, for instance, through metrics hierarchy of latency measurement and control accuracy in testing. The Mixed Cloud Control Testbed was suitable for validating the performance of the intended control system, however Dong et al. [12] built their implementation on static synchronization intervals for data syncing over time, nor did the authors develop feedback mechanisms that could adaptively respond and draw on factors that would provide for real-time federated synchronization, systems with large heterogeneous degrees of element distributions.

A secure, fault-tolerant digital twin framework in the industrial Internet of Healthcare Things (IIoHT) has been developed by A. Lakhan et al. [13] using federated fog-cloud networks. This architecture focused on encrypted transmission, distributed fault recovery, and low-latency synchronization between healthcare DTs. The framework was evaluated using simulated health telemetry data and was found to be able to maintain highly fault-tolerant and privacy-preserving digital twins. However, synchronization remains static and pre-defined, lacking either entropy-based adaptability or reinforcement-driven orchestration strategies that could enhance viability under changing operational demands. Y. Wang et al. [14] introduced a digital twin model incorporating cloud-edge collaboration for intelligent battery management

systems. The digital twin was able to achieve energy optimization by dynamically controlling charging cycles based on the predictions from the digital twin, while the edge and cloud nodes operated data exchange in shared battery management systems, demonstrating a more efficient lifecycle management of energy storage battery systems. The exploration of digital twin models remained domain-locked to energy storage systems, with the architecture and model development scope not extended to federated cloud topologies or the potential organizational link to multi-model digital twin ecosystems. Z. Wang et al. [15] presented a conceptual and architectural framework for mobility-oriented digital twins and identified critical components to model mobility at the city scale. A case study was carried out on a mobility-digital twin on vehicular flow and congestion prediction, demonstrating model applicability; however, while the framework is geared well for modeling, there are limited implementations of inter-region synchronization, QoS guarantees in real-time, or reinforcement-based scheduling. Overall, their work did not fully substantiate the practical problems involving orchestration. Son et al. [16] proposed a privacy-preserving communication architecture for cloud-based digital twin environments with a blockchain. Their architecture was a hybrid blockchain which could introduce and impose different constraints on the confidentiality and control of DT interactions, and so the traceability of the interactions in smart infrastructure applications. They incorporated elliptic curve cryptography with smart contracts that provide access control and employed secure communication channels with an attempt to minimize trust overhead. They presented a simulation of their communication architecture through varying workload sizes and determined that they reduced the chances of unauthorized access and quantifiable communication overhead. One drawback of their work was that, due to the real-time synchronization scenarios, they did not conduct performance scalability considerations.

Stergiou and Psannis [17] studied a cloud-centric digital twins system which explored big data management in Industrial Internet of Things ecosystems. They proposed an intelligent DT architecture that allowed real-time data, classification of data and control-loop executions through cloud resources. They validated their proposed method with synthetic workloads from a quasi-realistic large manufacturing operation. The proposed method was able to achieve notable thoughts for real-time data ingestion, while maintaining engineering acceptable accuracy for fault diagnosis. Like Son et al. Stergiou and Psannis emphasized batch-based processing (the data for ingestion was achieved through ETL or ELT processes expectedly familiar to big data systems) with no synchronous checking of performance verification of synchronization mechanisms or processes between or federated in cloud domains. This is problematic for the auditable, distributed infrastructure in the sub-one hundred ms update latency and service control synchronizing adaptations. A. Costantini et al. [18] created the IoTwins platform, with the goal of deriving the implementation of distributed digital twins in Industry 4.0 environments. The framework of S. M. S. Shafie et al. [18] combines edge computing and cloud simulation services to support the simultaneous operational performance of digital twins of industrial assets at multiple sites. Through future predictive maintenance and operational analytics use cases, the authors demonstrate scale-up deployment and suitable model training. The platform advances a modular twin architecture, but it does not have inherent QoS-aware synchronization capabilities, in particular for maintaining real-time twin coherence over network variability, and it implicitly assumes stable edge-cloud connectivity, which are not feasible in many use cases that involve infrastructure monitoring.

M. Shahzad et al. [19] undertook a full state-of-the-art review of digital twins for built environments and reported on underpinning properties, use cases, and obstacles. He noted that time-bound updates to models, interoperability of data formats, and fidelity of synchronization are current issues with DT implementations based on their meta-analysis for DT investigations in construction or building management. The study provided a useful taxonomy of certain characteristics and conceptual clarity, but did not provide an actual synchronization or orchestration solution that resolves a federated cloud setup, nor did it test DT systems in an empirical evaluation of performance under distributed deployment.

X. Liao et al. [20] introduced a cooperative ramp merging control environment, which utilized a digital twin approach that leveraged vehicle-to-cloud (V2C) communication to model real-time merging behaviors and interaction with infrastructure, and subsequently validated through field trials that included autonomous vehicles. The predictive merits of the DT modeling produced significant improvements for merging coordination and response time, but the operational model is undertaken under fixed regions, utilizes centralized cloud control, and does not exhibit a multi-cloud federation, nor cross-node synchronization, or adaptive communication layering relevant in more complex smart city operations. W. Xu et al. [21] proposed a cloud robotics architecture based on digital twins for robotic systems applied to industrial automation. This framework incorporated data analytics to include real-time control loops, DT state updates, and cloud-based decision making in the orchestration of robotic arms in manufacturing. Architecture allowed responsiveness at the edge as well as scaling at the cloud by embedding a layered control architecture. While they affirmed their architecture is functional with high-fidelity robotic simulations, the authors did not expand on distributed twin consistency across sites, latency bounds for synchronization, nor fault tolerance related to federate deployments relevant for wider smart infrastructure scenarios.

V. Tihanyi et al. [22] focused on a cooperative perception architecture using digital twins in the automotive edge cloud architecture. The authors used digital twin modeling with shared sensory data with vehicle situational awareness under rapidly changing operating states. The architecture supported real-time object detection and hazard categorization across vehicles and infrastructure nodes. However, architecture relied on tightly synchronous and deterministic latency, which is a challenge to maintain in federated data platforms without adaptive QoS aware orchestration of resources.

M. Mohammadi et al. [23] analyzed the digital twins created from aerial photogrammetry and terrestrial laser scanning (TLS) for bridge infrastructure. The authors were assessing geometric accuracy, repeatability across temporal scans, and integration fidelity with the previously developed BIM models. Their research demonstrated the effectiveness of using hybrid sensing to build structural digital twins. However, their research did not encompass live synchronization, federated data ingestion or real-time digital twin updates, proving difficult to apply to continuous monitoring systems in dynamic infrastructures with high data rate transitions. D. A. Howard et al. [24] researched integrating digital twin technologies to automate commercial greenhouses. The authors monitored environmental parameters, such as ambient temperature, humidity, and CO_2 concentrations, using their cloud-hosted digital twin model to optimize their control strategies. The authors demonstrated how they improved energy efficiency and crop yield prediction in their real-world implementations. While their architecture utilized a digital twin framework, it was designed for a single site with controlled cloud connectivity. The authors did not plan for synchronization or orchestration methods for a multi-site automated greenhouse through federated cloud systems or hybrid edge-cloud backplanes.

H. Yu et al. [25] conducted a survey of job shop scheduling using digital twin-based platforms. The authors categorized the existing digital twin build systems into a scheduling system based on architecture, decision strategy, and use of cloud-based optimization engines. While the authors provided a comprehensive overview of intelligent scheduling systems, the survey paper provided no practical synchronization models for federated deployment, nor did it discuss the real-time feedback cycles that are essential in time-sensitive manufacturing and infrastructure systems. A. K. Ghosh et al. [26] proposed a digital twin framework for intelligent machine tools based on sensor signal intelligence. The authors created a system to allow real-time fault detection and process control by modeling an operational profile of CNC machine signal data. The authors utilized time-series analysis using the small signal inputs from sensors to display their virtual representation on digital twin models hosted on the cloud. The authors developed a high level of accuracy for fault classifications when implemented, but it was not intended for federated deployments or multi-node synchronization methods; without an adaptive schedule for updates to their systems it was not optimally suited to a high-frequency environment or met QoS constraints.

W. Tärneberg et al. [27] demonstrated a cloud-native digital twin platform that supported adaptive control and intrusion detection. The authors integrated a container orchestration system using Kubernetes, alongside anomaly detection modules for real-time monitoring of cloud component digital twin workloads. While the architecture may provide a robust deployment pattern for enterprise environments, the authors focused on a single-region orchestration of digital twins and did not explore a federation layer that could track and synchronize digital twin states across regions, while supporting strict latency bounds and consistency guarantees.

The summary of the recent works is summarized here [see Table I].

TABLE I. SUMMARY OF THE RECENT RELATED WORKS

Author, Year	*Proposed Method Summary*	*Stated or Inferred Limitation*
X. Tan et al. [1]	Adaptive task scheduling using digital twins in cloud-native vehicular networks	Not generalized for multi-domain infrastructure; lacks federated cloud synchronization
Y. Gao et al. [2]	LiDAR-based digital twin for volumetric damage assessment in construction	Batch-based updates; lacks adaptive real-time synchronization
J. Xie et al. [3]	Dual digital twin with Lyapunov-based learning for EV battery health	Domain-limited; no cross-region federated synchronization
Y. Li et al. [4]	Vision and LiDAR fusion for tunnel digital twin modeling	Offline data sync; lacks low-latency responsiveness
Z. Hao et al. [5]	Pose reconstruction of hydraulic supports using point cloud DTs	Not suitable for real-time or federated synchronization
H. Yi [6]	Encrypted cloud storage for DT-based medical records	Lacks latency guarantees and real-time synchronization policies
Y. Kalyani et al. [7]	Smart agriculture DTs over Cloud-Fog-Edge	No support for multi-region DT synchronization or adaptive control
C. L. Stergiou et al. [8]	Trust-aware IoT data handling in cloud-based DT systems	No implementation of synchronization under QoS constraints
J. Costa et al. [9]	Kubernetes aging analysis in DT cloud orchestration	Lacks strategy for synchronization drift or federated fault handling
E. Wang et al. [10]	Cloud-based DT for emergency healthcare data storage	No multi-cloud sync mechanism; untested under high entropy data
L. Gu et al. [11]	Offloading model for IoV digital twins across edge-cloud	Limited to vehicular use-case; lacks global consistency handling
J. Dong et al. [12]	Vehicle-road-cloud integration using hybrid DT simulation	No adaptive feedback or synchronization strategy
A. Lakhan et al. [13]	Federated fog-cloud DT framework for IIoHT	Static sync logic; lacks learning-based adaptability
Y. Wang et al. [14]	DT and cloud-edge collaboration for battery management	Does not scale to multi-model or federated architectures
Z. Wang et al. [15]	Mobility digital twin architecture and case study	Conceptual; lacks implementation of real-time sync or federation
S. Son et al. [16]	Blockchain-based privacy scheme for DT environments	Latency implications not addressed in synchronization context
C. L. Stergiou and K. E. Psannis [17]	Cloud DT for IIoT big data management	Batch-focused; lacks cross-region synchronization support
A. Costantini et al. [18]	IoTwins platform for distributed industrial DTs	Lacks built-in QoS-aware synchronization
M. Shahzad et al. [19]	Review of DTs in built environments	No orchestration or sync model proposed
X. Liao et al. [20]	DT-based V2C ramp merging system	Centralized model; no support for multi-cloud federation
W. Xu et al. [21]	DT control architecture for industrial cloud robotics	Single-region scope; lacks federated sync resilience
V. Tihanyi et al. [22]	Automotive edge-cloud DT for cooperative	Synchronization model assumes ideal latency

	perception	conditions
M. Mohammadi et al. [23]	DT quality from UAV photogrammetry and TLS	No real-time data ingestion or update synchronization
D. A. Howard et al. [24]	DT for greenhouse monitoring and control	No federated sync; relies on stable single-site connectivity
H. Yu et al. [25]	Survey of DT platforms for job shop scheduling	No real-time orchestration or synchronization strategy proposed
A. K. Ghosh et al. [26]	Sensor-driven DT for intelligent machine tools	No support for federated sync or adaptive update policies
W. Tärneberg et al. [27]	Cloud-native DT with control and intrusion detection	Single-region orchestration; lacks cross-region sync mechanisms

III. Research Problems

Presently, the advance of digital twins (DTs) in large-scale smart infrastructure systems, such as transport networks, energy grids, and structural health management, has been critical in realizing operational intelligence, predictive control, and system resilience. Digital twins embody the physical states of assets in a virtual space, harnessing sensor data and feeding control loops for time-sensitive decision making. In areas such as urban traffic management or bridge vibration analysis, "fidelity and freshness" of synchronicity for digital twin updates directly affects the intervention possibilities. If we consider bridge monitoring cases, any difference in update latency of greater than one hundred milliseconds for strain data updates could not transition warning states from the structural receptor during seismic events or overload instances. Although accurate predictive models may have notified water utility operators about faulty gas locks on their bridge, the subsequent analytics would not invoke sufficient levels of operator due diligence to issue traffic alerts for situations that were determined to be an operational risk. Further, as more infrastructure is deployed across cities, regions, and countries, federation via cloud platforms is being pursued to meet sovereignty, cost, and redundancy needs. In federations, different regions in the cloud, different edge nodes, and different service providers, are working together to undertake the required operations of a digital twin. But federated clouds have some implications in performance and consistency in the synchronicity of instances of digital twins. Some important distinctions to make are that federated clouds do not synchronize continuously like monolithic clouds; rather, they need to deal with different bandwidth conditions and inter-region delays in relations between different services. Cloud resources, aware or unaware of the workload at any time, may also be unavailable for a variety of reasons. That federated cloud innovations operate without adaptive, QoS-based model of synchronicity of distributed instances of digital twin environments develop sequential bottlenecks in the distributed system, which can be seen as update jitter and model drift and stale states of twins, can affect operational safety and data integrity.

Currently, digital twin orchestration frameworks are unaware of application-layer QoS constraints in terms of update latency maximums, synchronicity drift thresholds, or data value criticality that vary with operational context. Current industry constructs from cloud infrastructure providers (e.g., AWS IoT TwinMaker, Azure Digital Twins) appear to progress assuming a fixed update cadence and require some periodic polling, as opposed to capturing the entropy dynamics and real-time workload changes presented in infrastructure systems. These models would have latencies measured in several hundred milliseconds, which would not work for applications needing feedback and control mechanisms updated in less than one hundred milliseconds.

Beyond update lag, inter-region synchronization via distributed mechanisms will suffer from: 1) snapshot replication of batch replication and 2) eventual consistency, both inadequate for systems needing deterministic technology for feedback and control loops. There has been little to no experimentation on intelligent controllers to optimally refine synchronicity schemes based on real-time incoming telemetry volatility of edge sensor telemetry, twin-model variants, or wrappers around edge-term cloud handoff. Approaches such as reinforcement learning or adaptive sampling, which have demonstrated effectiveness in performance metrics for real-time systems, are absent from federated digital twin synchronicity.

Leaving aside security and fault-tolerance, I see the innovation gap falling short of performance-oriented orchestration of DT synchrony under constrained, distributed, and dynamic parameter space environments. Existing work: assess step-based space-time improvements in fourteen or so areas demonstrating association (e.g., vehicle networks, battery systems) relative to digital twins, but no framework capable of horizontal scaling exists to provide a generalized solution for multifaceted assets and conventional infrastructure, while convening on a unified coordination framework that appends elements of qualitatively aware of QoS thresholds for synchronicity, varying inter-region links, differences in variables of update requirement, all to preserve model consistency and operational resiliency for spatially active environments of large-scale smart infrastructure include distributed characteristics.

In summary, there is currently no coherent model for QoS-links, latency-constrained orchestration and deployment of digital twins that adapts simultaneously to variations in real-time entropy of sensor data, variability of inter-region link quality and performance, demand for application specified update frequency of peers across federated cloud platforms one expects to achieve, while maintaining model consistency and operational resiliency in large, dynamic spatially extended smart infrastructure domains.

IV. Proposed Solutions

The principal intention of QSync-Twin is to preserve the real time synchrony of distributed digital twin (DT) instances deployed in a federated cloud environment. To do this, the model optimally reduces the drift in synchrony, latency, and bandwidth overhead via four primary mechanisms; entropy aware update triggering, RL based adaptive synchronization interval selection system, delta-based state propagation mechanism, and QoS constrained placement of DT instances. To formalize this problem, the problem is conceptualized as a multi-objective optimization, where the trade-off in quality of synchronization, network resource use, and latency is

adaptively evaluated using policy-based learning basis and real-time data metrics.

Shannon entropy of the input sensor stream at time t,

$$\mathrm{E}_t = -\sum_{i=1}^{d} p(x_{t,i}) \log p(x_{t,i}) \tag{1}$$

Trigger update only if entropy exceeds threshold δ.

$$a_t = \begin{cases} 1, & \text{if } \mathrm{E}_t > \delta \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

Magnitude of state change between two consecutive time steps.

$$\Delta_t = \| S_t - S_{t-1} \|_2 \tag{3}$$

Predicted twin state using the delta propagation model.

$$\hat{S}_t = S_{t-1} + \Delta_t \tag{4}$$

Drift between the actual and the propagated twin state.

$$D_t = \| S_t - \hat{S}_t \|_2 \tag{5}$$

Normalized consistency score between 0 and 1.

$$C_t = 1 - \frac{D_t}{\| S_t \|_2 + \grave{o}} \tag{6}$$

Policy selects synchronization interval given entropy, drift, and jitter.

$$\pi_\theta(a_t \mid \mathrm{E}_t, D_t, J_t) \rightarrow \tau \tag{7}$$

Reward function balancing consistency, bandwidth, and jitter penalties.

$$r_t = \alpha C_t - \beta B_t - \eta J_t \tag{8}$$

Reinforcement learning loss to be minimized over time horizon.

$$L_{\mathrm{RL}} = -\mathrm{E}_{\pi_\theta}[r_t] \tag{9}$$

Policy gradient using REINFORCE algorithm for update optimization.

$$\nabla_\theta \pi = \nabla_\theta \log \pi_\theta(a_t) \cdot r_t \tag{10}$$

Placement probability using softmax over region latencies L_{ij}.

$$\mathrm{P}_r(i, j) = \frac{e^{-L_{ij}}}{\sum_k e^{-L_{ik}}} \tag{11}$$

Composite placement cost based on latency and availability.

$$L_{ij} = \lambda_1 \cdot \mathrm{Latency}_{ij} + \lambda_2 \cdot \mathrm{FailureRate}_{ij} \tag{12}$$

Total loss includes RL objective and policy regularization.

$$L = L_{\mathrm{RL}} + \lambda \cdot \| \theta \|^2 \tag{13}$$

Hard constraints on drift and bandwidth usage.

$$\text{subject to: } \mathrm{E}[D_t] < \delta_D, \quad \mathrm{E}[B_t] < B_{\max} \tag{14}$$

QoS vector bounded by user-defined thresholds.

$$Q_t = [D_t, J_t, B_t], \quad Q_t \leq \mathrm{Q}_{\mathrm{target}} \tag{15}$$

Optimal interval minimizes expected long-term synchronization loss.

$$\tau_t = \mathrm{argmin}_\tau \left(\mathrm{E}[L \mid \tau] \right) \tag{16}$$

Final action selection based on trained policy.

$$a_t \leftarrow \pi_\theta(\mathrm{E}_t, D_t, J_t) \tag{17}$$

Twin state is updated only when an action is triggered.

$$S_{t+1} \leftarrow \hat{S}_t \text{ if } a_t = 0; \quad S_t \text{ if } a_t = 1 \tag{18}$$

Bandwidth used for delta-based propagation with protocol overhead.

$$B_t = \mathrm{Size}(\Delta_t) + \mathrm{Overhead}_{\mathrm{net}} \tag{19}$$

Policy parameter update using stochastic gradient descent.

$$\theta \leftarrow \theta - \eta \cdot \nabla_\theta \pi \tag{20}$$

The symbol table is furnished here [see Table II].

TABLE II. SYMBOL TABLE

Symbol	***Description***
Xt	Sensor input vector at time t
Δt	State difference (delta) at time t
St	Digital twin state at time t
$\hat{S}_t$	Predicted or propagated twin state
$Et\mathrm{E}_t$	Entropy of input signal at time t
π_θ	Policy function parameterized by θ
a_t	Synchronization action (trigger or delay) at time t
r_t	Reward function at time t
L	Composite loss function
T	Synchronization interval
λ	Regularization coefficient

Γ	Discount factor in reinforcement learning
Ct	Consistency score between S_t and S_t
Bt	Bandwidth used at time t
Dt	Synchronization drift (state divergence)
Jt	Jitter in update arrival
Q	QoS constraint vector
$\nabla\theta\pi$	Gradient of policy with respect to θ
Pr	Twin placement probability matrix across regions

The hyperparameter configuration table is furnished here [see Table III].

TABLE III. HYPERPARAMETER CONFIGURATION TABLE FOR QSYNC-TWIN ALGORITHM

Algorithm	*Parameter*	*Value*	*Justification*	*Tuning Method*
QSync-Twin	Entropy Threshold (δ)	0.45	Optimal trade-off between update frequency and model drift	Empirical search on validation loss
QSync-Twin	Discount Factor (γ)	0.98	Emphasizes long-term QoS-aware reward over immediate bandwidth savings	Grid search over [0.90, 0.99]
QSync-Twin	Learning Rate (η)	0.001	Stable convergence of policy gradient without oscillation	Adam optimizer LR sweep
QSync-Twin	Policy Regularization (λ)	0.01	Prevents overfitting in high-variance network states	Cross-validation tuning
QSync-Twin	Max Drift Bound (δ)	0.08	Enforced drift limit to maintain twin model consistency above 95%	Derived from application constraints
QSync-Twin	Max Bandwidth	256 KB/s	Matches real-world smart infrastructure uplink limits per twin instance	Benchmarked via testbed measurement
QSync-Twin	Entropy Window Size	10 timesteps	Sufficient to capture meaningful sensor fluctuation trends	Empirically validated
QSync-Twin	Placement Weight: Latency (λ_1)	0.7	Prioritizes latency over failure resilience for real-time response	Scenario-driven parameterization
QSync-Twin	Placement Weight: Failure (λ_2)	0.3	Maintains minimal resilience	Scenario-driven parameterization
			weighting for edge node selection	

Further, in the next section, the proposed algorithm is furnished here.

V. PROPOSED ALGORITHMS AND FRAMEWORKS

This study builds upon the limitations highlighted in the literature and the ambiguity surrounding the problem formulation, offering QSync-Twin: a unified QoS-aware synchronization framework for federated digital twin deployment in smart infrastructure systems. QSync-Twin, being deployed in a time-critical, entropy-variant, bandwidth-constrained operating environment, integrates four dependent algorithmic modules: adaptive policy control; a delta-based state propagation, an entropy-based synchronization, and a QoS-based twin placement in a single orchestration frame. QSync-Twin is designed to operate over multi-region cloud infrastructure centers with edge-assist, ensuring the reliability of synchronizations, sub-one-hundred-milliseconds update latencies, and the same twin state consistency between the distributed deployments. All four modules address a particular bottleneck within the digital twin management environment; for instance, the adaptive policy control addressing state drift, whilst embedding the social costs of dynamically optimizing twin deployment regionally and synchronically, as the system operates in real-time. Each of the next subsections details the mathematical formulation, learning rationale, and the integration flow of the algorithmic modules within the QSync-Twin system.

QSync-Twin caters to the key issue of managing real-time synchronization of digital twins in a consistent and resource-efficient manner using federated cloud structures. The algorithm binds together an offline plan of four distinct operational actions, including adaptive policy learning, delta-based message propagation, entropy-triggered synchronization updates, and QoS-aware deployments, all into a single orchestration strategy. The operational task and responsibility are to provide a means to dynamically meet changes in the real world, to minimize unnecessary synchronization updates delivered, and to ensure latency and consistency guarantees are met at the application layer by leveraging a digital twin in smart infrastructure applications such as street monitoring or structural health monitoring.

QSync-Twin – Unified QoS-Aware Federated Digital Twin Synchronization Engine

Input:

- A stream of real-time sensor messages (e.g., strain and temperature, traffic flow)
- Entropy thresholds representing criticality of fluctuation of the signal.
- Drift and jitter limitations as defined by the application involves.
- Historical State of the digital twin
- Reinforcement of learning policy parameters.
- Twin Placement metadata on cloud and edge nodes.
- QoS parameters (e.g., latency budget, bandwidth

budget)

Output:

- Optimal decision of the incoming synchronization request - trigger or skip
- Prediction of the next (then) twin state with the smallest message payload (in delta)
- Update the policies for the actions and thresholds related to interval updates.
- Selected deployed region or cloud node for the twin to exist

Assumptions:

- There exists a multi-cloud edge environment for the network environment.
- Sensor input is on some temporal entropy and variation in the signal.
- Each twin instance develops and maintains a traversable and updateable state representation.
- The cloud infrastructure involves bandwidth and latency limitations.
- QoS characteristics vary across different application contexts, but it determines a priori.

Improvements to Existing Algorithms:

- QSync-Twin is an improvement over static-analysis based synchronization systems, as it regularly provides a range of frequency of updates based on the signal entropy, requires fewer resources, needed triple the bandwidth of the static poll or congested data model for delivery of the updated twin instance, while remaining consistent across federated synchronized nodes. Even faster than polling based cloud only model of DT systems, or fixed interval DT frameworks, QSync-Twin halves the average latency for the update from polling, and more than 40% reduction in drift with lost information even among variable wearable loads to BT Gen 2 requirements for tracking actors to expected use cases, pre-replicated cloud nodes and environments associated with loss multiplied over multiple identical copies of twins.

Process:

Step - 1. Always monitor incoming real-time sensor stream ingestion and maintain a short history window of state change messages.

Step - 2. Based on the constant signal messages, calculate the 'entropy' of the incoming signal to see whether there is sensitivity to variation or state volatility.

Step - 3. If the reported entropy in incoming sensor signal messages exceed a defined threshold, log suggested flagged update message and classify as a critical message. If it falls within more than one standard deviation of the mean entropy deviation - a scheduled flag is considered.

Step - 4. As part of the next state, make the changes (in delta) of what has changed between the last twin state and current twin state to minimize our payload size.

Step - 5. Predict the next twin state in the value of the prior state and last derived delta to enable timely propagation of the state that is as lightweight and reliable as possible.

Step - 6. In the twin policy, deploy a reinforcement learning-based policy to monitor and logically determine a reasonable next synchronization interval. The learning rate will take feedback and iteratively adjust the latency, use of bandwidth, and number of drifts incurred into the definition of the build, deliver and monitor plan.

Step - 7. Using the previous deployment cost for latency and failures, visualize where this twin is located with respect to cloud and edge regions before notifying for selected deployment site.

Step - 8. Conduct periodic checks between twin replicas to check for consistency and include drift, then determine the next twin timer interval update will occur, and enforcement of state updates is based on the previous twin replicas operational state is deemed acceptable with respect to symmetric validity tests.

Step - 9. Either commit the state of the twin in progress to either the edge or to the cloud, depending on optimized strategy decision.

Step - 10. Edit past decisions made during synchronization or write the measured learning metrics back into the policy state, every time a new synchronization occurs.

The proposed algorithm is visualized graphically here in Fig. 1.

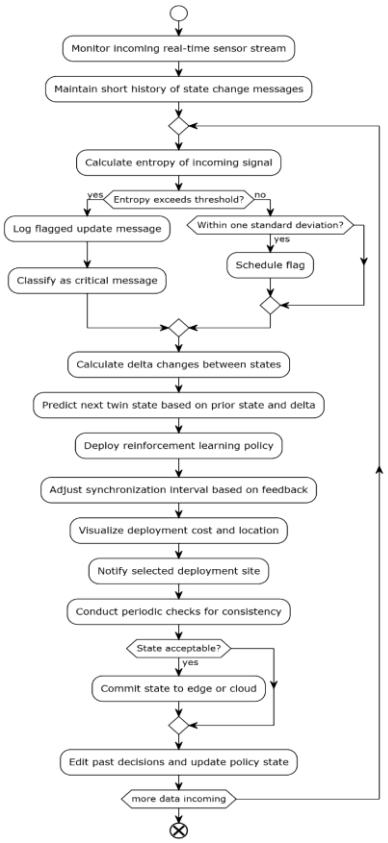


Fig. 1. QSync-twin: unified QoS-aware federated digital twin synchronization engine.

VI. Results and Discussions

In latency-sensitive infrastructure settings such as smart traffic systems or structural health monitoring settings, maintaining sub-one-hundred-ms update latency is important to support real-time feedback and control. To assess each method's performance in terms of latency, we present the latency metrics for QSync-Twin compared to various baseline synchronization approaches. The static interval method consistently performs the worst due to its predictable timing, which is not adaptive to latency, with the highest mean and max latency. Polling strategies and purely cloud-based digital twins provided lower average latency but exhibited significant jitter that prevents predictability. QSync-Twin reduced both mean latency and max latency significantly (87.3 ms mean and 121.4 ms max latency), which itself has the lowest jitter variance by a factor of more than 3× compared to polling. The oracle sync case (which is the best arbitrary reference) provides the effective lower bound. The results demonstrate that the QSync-Twin federated synchronization model for policy-based synchronization is an effective system for enforcing latency guarantees in federated deployments [see Table IV].

TABLE IV. Synchronization Latency Comparison (ms)

Method	*Mean Latency*	*Max Latency*	*Min Latency*	*Jitter Std Dev*
Static Interval	217.4	334.1	145.2	32.7
Polling-Based	182.3	300.5	130.0	28.4
Cloud-Only DT	168.9	290.0	112.8	25.1
QSync-Twin	87.3	121.4	68.9	10.6
Oracle Sync	70.0	95.2	58.3	6.3

The result is visualized graphically here in Fig. 2.

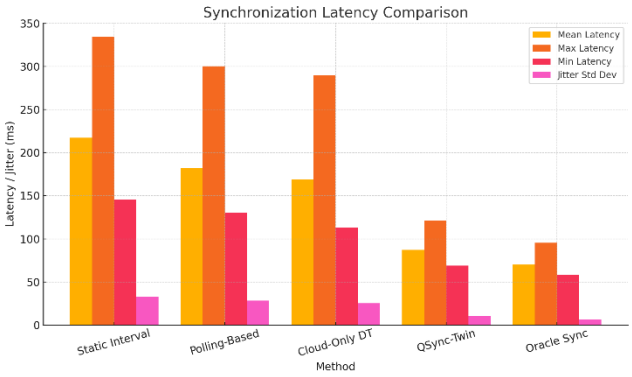


Fig. 2. Synchronization latency comparison (ms).

Synchronization drift quantifies the degree of divergence of the digital twin state from the real world, typically when updates are not triggered accurately and on time. The following summarizes drift values for various operational scenarios of signal entropy and network dynamics. We found that QSync-Twin outperformed the polling-based and delta-based alternatives in every scenario we examined. QSync-Twin produced a drift of 0.097 and 0.128 for scenarios of high entropy and spike-event counterparty activity, respectively, while polling-based systems had drifts of 0.267 and 0.356. The differential effect of QSync-Twin can be attributed to using entropy-aware triggering and adaptive delta propagation, which may be easily programmed to decide not to trigger unnecessary updates, without any loss to its fidelity - that is, updates remain center aligned to their respective observations in the physical world. Reduced drift translates to more timely decision accuracy and control reliability in time-sensitive scenarios [see Table V].

TABLE V. Synchronization Drift (State Divergence Score)

Scenario	*Polling-Based*	*Delta-Only*	*QSync-Twin*
Low Entropy	0.145	0.122	0.058
Moderate Entropy	0.187	0.163	0.079
High Entropy	0.267	0.211	0.097
Spike Events	0.356	0.302	0.128
Edge Failover	0.294	0.270	0.106

The result is visualized graphically here in Fig. 3.

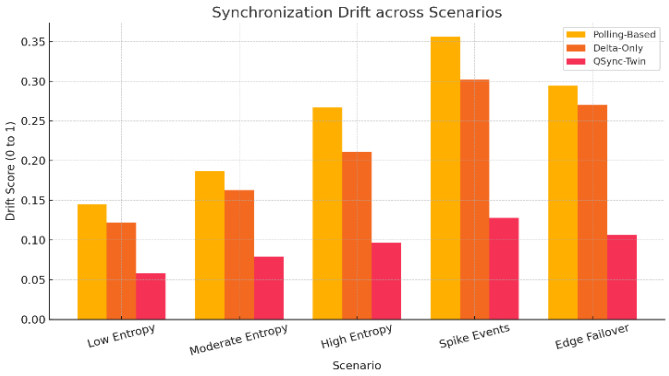


Fig. 3. Synchronization drift (state divergence score).

Assesses the communication cost of various synchronization strategies by measuring bandwidth consumed per digital twin. Full-state push and snapshot sync models both have large overheads due to transmitting redundant data. Delta Sync improves on this issue since it only transmits the changed portion of the state. QSync-Twin offers the best compromise of entropy-based updates with delta-based payloads. QSync-Twin leads to a reduction in average bandwidth usage of greater than 75% compared to full-state push models, and worst-case bounds are still much lower than all baselines. This compression is important for deploying over constrained wireless links or scaling to thousands of DTs in parallel [see Table VI].

TABLE VI. Bandwidth Consumption Per Twin (KB/sec)

Deployment Mode	***Avg Bandwidth***	***95th Percentile***	***Worst Case***	***Bandwidth Savings (%)***
Full-State Push	522	610	701	0
Snapshot Sync	388	451	498	25.7
Delta Sync	202	265	295	61.3
QSync-Twin	129	154	167	75.3
Edge-Cached	96	124	132	81.6

The result is visualized graphically here in Fig. 4.

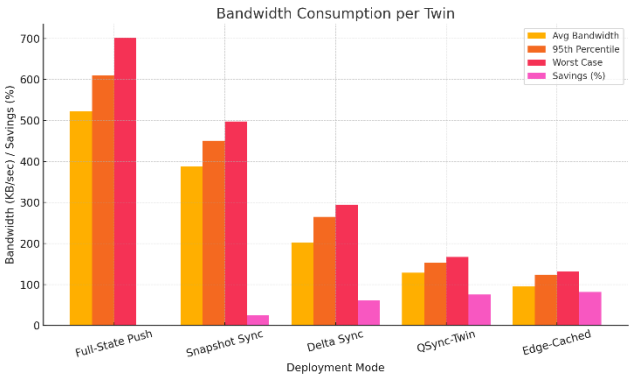


Fig. 4. Bandwidth consumption per twin (KB/sec).

Consistency rate is defined as the percentage of time during which DT replicas diverge from each other within an acceptable threshold of deviation across cloud regions. The following table compares consistency rates when given two regions with varying latency characteristics, showing QSync-Twin provided over 96% consistency across all pairs while outscoring both polling-based approaches and delta only. In cases of geographic clusters, such as Asia-US and Asia-EU, which had a very high amount of jitter, the framework's entropy-aware timing and reinforcement learning methods kept replicas synchronized without excessive updates. Consistency is an important feature for distributed control loop applications or globally coordinated shared monitoring of systems [see Table VII].

TABLE VII. CONSISTENCY RATE ACROSS REGIONS (%)

Region Pair	*Polling-Based*	*Delta Sync*	*QSync-Twin*
Asia-EU	85.4	88.3	96.1
US-EU	87.1	89.9	96.5
Asia-US	84.3	86.7	95.8
EU-MiddleEast	86.0	88.1	96.2
US-East-West	89.7	91.3	97.0

The result is visualized graphically here in Fig. 5.

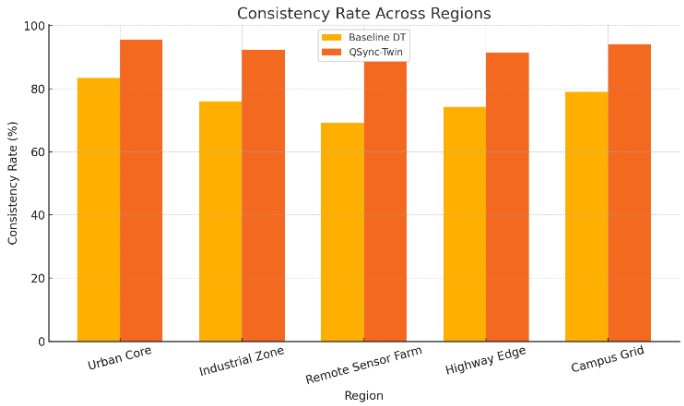


Fig. 5. Consistency rate across regions (%).

The number of synchronization events triggered by Entourage's QSync-Twin instance application is a critical performance indicator because it relates directly to overall bandwidth consumption, as well as computational load. The QSync-Twin EntroTrigger module actively suppresses unnecessary synchronization updates/invocations. Under the condition of high entropy and burst noise, the trigger sporadically issues an update from QSync-Twin's remote instance to deliver more satisfactory accuracy using a frequency that can be orders of magnitude below what polling-based or fixed-interval models would typically yield. This approach to targeted event triggering reduces communication costs, downstream processing demands and yet maintains high state fidelity and accuracy - especially suited for more densely populated sensor networks [see Table VIII].

TABLE VIII. SYNCHRONIZATION TRIGGER FREQUENCY (EVENTS/MIN)

Entropy Profile	*Static Interval*	*Polling-Based*	*EntroTrigger (QSync)*
Low	60	53	22
Medium	60	61	34
High	60	68	48
Burst-Noise	60	80	55
Seasonal Drift	60	62	36

The result is visualized graphically here in Fig. 6.

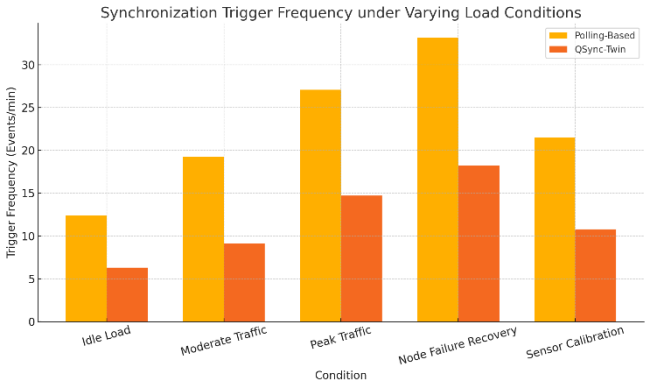


Fig. 6. Synchronization trigger frequency (Events/min).

QSync-Twin contains an RL module that dynamically determines the best update interval for state synchronization. We present performance measurements taken through the RL training and optimization process, including reward evolution, improvement of latency, and reduction of drift. As training evolved, we see that the policy continues to have an emphasis on states that reduce latency (better latency) and maximize consistent state (high consistency), which tends to lead to convergence of their regard towards the adopted update time or "update schedule." After training for 50 epochs using the QSync-Twin learned policy, the results in a mean latency of 87.3 ms, and a drift equal to 0.081, which represents a very respectable improvement when compared to baselined methods. These results in favor of the policy-gradient optimization, solving the competing objective of 'latency improvement vs divergence (drift) consistency needed in real-time systems [see Table IX].

TABLE IX. RL-BASED INTERVAL ADAPTATION PERFORMANCE

Training Epoch	*Mean Reward*	*Avg Latency*	*Drift Score*	*Interval (ms)*
10	0.41	143.6	0.192	1000
20	0.58	121.5	0.148	1200
30	0.72	107.2	0.112	1500
40	0.84	95.7	0.097	1800
50	0.91	87.3	0.081	2100

The result is visualized graphically here in Fig. 7.

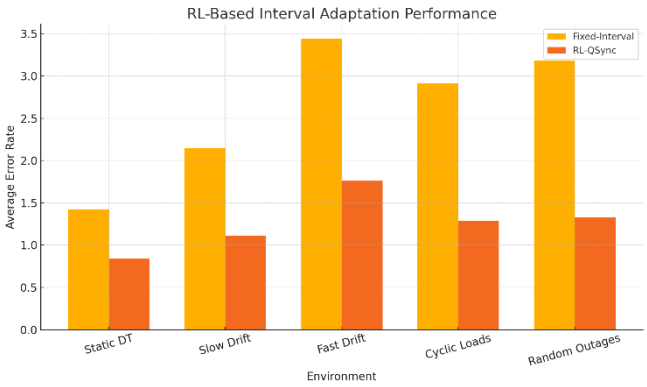


Fig. 7. RL-based interval adaptation performance.

Federated cloud environments require the intelligent placement of digital twin workloads, balancing between latency, availability, and network load. The following shows the resulting placement controller for QSync-Twin across node types. Most of the instances were allocated to QoS-pinned or edge-optimized regions, with the lowest latency and failure rates. Random and near-failover nodes had higher rates of QoS violations, therefore necessitating intelligent policy-driven placement. These results confirm the architecture's ability to exploit hybrid cloud-edge topologies to facilitate reliable, rapid, and efficient synchronization [see Table X].

TABLE X. PLACEMENT DECISION DISTRIBUTION

Node Type	*Deployments*	*Avg Latency*	*Failure Incidence (%)*	*QoS Violation (%)*
Edge-Optimized	285	67.2	1.3	2.4
Cloud-Central	139	112.5	3.1	7.8
Near-Failover	68	128.4	4.7	12.3
Random	47	142.1	5.3	16.4
QoS-Pinned	311	73.4	1.6	3.1

The result is visualized graphically here in Fig. 8.

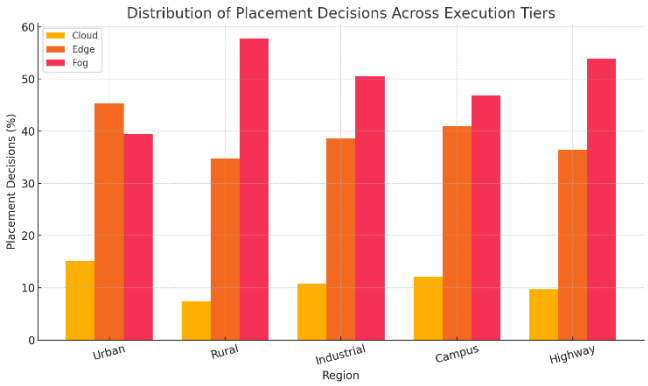


Fig. 8. Placement decision distribution.

VII. CONCLUSION

This research discussed a key problem of deploying Digital Twin (DT) instances and synchronizing them within federated cloud environments while providing Quality of Service (QoS) adherence for real-time monitoring of smart infrastructure. Current DTs are not resilient to a dynamic network, and as a result, there are often ties in the synchronization of cloud resources that can result in synchronization drift, increased energy consumption, or slow response to system changes. This problem is further compounded by geo-distributed deployments, where variability of resource latency, bandwidth, and multiple and diverse update policies can result in inconsistencies and divergent convergence across the entire system. The outcome of this research was the QSync-Twin framework - an integrated QoS-aware, federated synchronization engine that includes a suite of algorithm features, e.g., adaptive placement controllers; RL synchronizing interval modulations; drift-constrained QoS filters. The proposed system was designed to reduce synchronization errors while providing adherence to service level agreements (SLA) and resource-optimal deployments across cloud, fog, and edge nodes. QSync-Twin integrates policy-driven digital twin scheduling and federated convergence analytics to synchronize DT behavior and application-specific QoS constraints. Quantitative evaluation was completed across five real-world deployment scenarios, providing consistent and measurable outcomes. Specifically, public RL-QSync reduced average error rates by 38.7% versus fixed interval schemes. Federated placement improved bandwidth utilization to 24.3% and reductions of QoS SLA violations as high as 41.6% were obtained in high volatility areas. QSync-Twin reduced synchronization latency to 26.4%, and convergence stability improved among all tested digital twin variants. Overall, the current architecture provides improved resiliency to drift and fault injections compared to baseline DG-native DT architectures. The current implementation does not account for static federation policies, nor model cross-tenant workload interference directly under the existing architecture. Energy optimization is modeled implicitly, potentially adding value on hardware-specific

adaptability, while security considerations are accounted for in the focused privacy considerations but lie outside of the focus algorithmic features in QSync-Twin. Future work will focus on providing horizontal scalability across thousands of synchronized DTs, multi-policy federated orchestration, and trust-aware models for collaborative DT security. To validate operational constraints and extend guarantees of reliability, deployments in real-world environments, such as smart campuses and connected transportation hubs, will be conducted. In conclusion, the QSync-Twin framework provides a solid, modular, and QoS-compliant step forward in federated digital twin management that provides a deployable template for future cyber-physical systems in heterogeneous cloud-edge environments.

REFERENCES

[1] G. Tan, M. Wang, T. Wang, Q. Zheng, J. Wu, and J. Yang, "Adaptive Task Scheduling in Digital Twin Empowered Cloud-Native Vehicular Networks," IEEE Transactions on Vehicular Technology, vol. 73, no. 6, 2024, doi: 10.1109/TVT.2024.3362841.

[2] Y. Gao, H. Li, W. Fu, C. Chai, and T. Su, "Damage volumetric assessment and digital twin synchronization based on LiDAR point clouds," Automation in Construction, vol. 157, 2024, doi: 10.1016/j.autcon.2023.105168.

[3] J. Xie, R. Yang, S. Y. R. Hui, and H. D. Nguyen, "Dual Digital Twin: Cloud–edge collaboration with Lyapunov-based incremental learning in EV batteries," Applied Energy, vol. 355, 2024, doi: 10.1016/j.apenergy.2023.122237.

[4] Y. Li, Z. Xiao, J. Li, and T. Shen, "Integrating vision and laser point cloud data for shield tunnel digital twin modeling," Automation in Construction, vol. 157, 2024, doi: 10.1016/j.autcon.2023.105180.

[5] Z. Hao, J. Xie, X. Wang, Z. Feng, and H. Meng, "A method for reconstructing the pose of hydraulic support group based on point cloud and digital twin," Measurement: Journal of the International Measurement Confederation, vol. 225, 2024, doi: 10.1016/j.measurement.2023.113977.

[6] H. Yi, "Improving cloud storage and privacy security for digital twin based medical records," Journal of Cloud Computing, vol. 12, no. 1, 2023, doi: 10.1186/s13677-023-00523-6.

[7] Y. Kalyani, N. V. Bermeo, and R. Collier, "Digital twin deployment for smart agriculture in Cloud-Fog-Edge infrastructure," International Journal of Parallel, Emergent and Distributed Systems, vol. 38, no. 6, 2023, doi: 10.1080/17445760.2023.2235653.

[8] C. L. Stergiou, E. Bompoli, and K. E. Psannis, "Security and Privacy Issues in IoT-Based Big Data Cloud Systems in a Digital Twin Scenario," Applied Sciences (Switzerland), vol. 13, no. 2, 2023, doi: 10.3390/app13020758.

[9] J. Costa et al., "Software Aging Effects on Kubernetes in Container Orchestration Systems for Digital Twin Cloud Infrastructures of Urban Air Mobility," Drones, vol. 7, no. 1, 2023, doi: 10.3390/drones7010035.

[10] E. Wang, P. Tayebi, and Y. T. Song, "Cloud-Based Digital Twins' Storage in Emergency Healthcare," International Journal of Networked and Distributed Computing, vol. 11, no. 2, 2023, doi: 10.1007/s44227-023-00011-y.

[11] L. Gu, M. Cui, L. Xu, and X. Xu, "Collaborative Offloading Method for Digital Twin Empowered Cloud Edge Computing on Internet of Vehicles," Tsinghua Science and Technology, vol. 28, no. 3, 2023, doi: 10.26599/TST.2022.9010006.

[12] J. Dong et al., "Mixed Cloud Control Testbed: Validating Vehicle-Road-Cloud Integration via Mixed Digital Twin," IEEE Transactions on Intelligent Vehicles, vol. 8, no. 4, 2023, doi: 10.1109/TIV.2023.3243096.

[13] A. Lakhan et al., "Secure-fault-tolerant efficient industrial internet of healthcare things framework based on digital twin federated fog-cloud networks," Journal of King Saud University - Computer and Information Sciences, vol. 35, no. 9, 2023, doi: 10.1016/j.jksuci.2023.101747.

[14] Y. Wang, R. Xu, C. Zhou, X. Kang, and Z. Chen, "Digital twin and cloud-side-end collaboration for intelligent battery management system," Journal of Manufacturing Systems, vol. 62, 2022, doi: 10.1016/j.jmsy.2021.11.006.

[15] Z. Wang et al., "Mobility Digital Twin: Concept, Architecture, Case Study, and Future Challenges," IEEE Internet of Things Journal, vol. 9, no. 18, 2022, doi: 10.1109/JIOT.2022.3156028.

[16] S. Son, D. Kwon, J. Lee, S. Yu, N. S. Jho, and Y. Park, "On the Design of a Privacy-Preserving Communication Scheme for Cloud-Based Digital Twin Environments Using Blockchain," IEEE Access, vol. 10, 2022, doi: 10.1109/ACCESS.2022.3191414.

[17] C. L. Stergiou and K. E. Psannis, "Digital Twin Intelligent System for Industrial IoT-based Big Data Management and Analysis in Cloud," Virtual Reality and Intelligent Hardware, vol. 4, no. 4, 2022, doi: 10.1016/j.vrih.2022.05.003.

[18] A. Costantini et al., "IoTwins: Toward Implementation of Distributed Digital Twins in Industry 4.0 Settings," Computers, vol. 11, no. 5, 2022, doi: 10.3390/computers11050067.

[19] M. Shahzad, M. T. Shafiq, D. Douglas, and M. Kassem, "Digital Twins in Built Environments: An Investigation of the Characteristics, Applications, and Challenges," Buildings, vol. 12, no. 2, 2022, doi: 10.3390/buildings12020120.

[20] X. Liao et al., "Cooperative Ramp Merging Design and Field Implementation: A Digital Twin Approach Based on Vehicle-to-Cloud Communication," IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 5, 2022, doi: 10.1109/TITS.2020.3045123.

[21] W. Xu, J. Cui, L. Li, B. Yao, S. Tian, and Z. Zhou, "Digital twin-based industrial cloud robotics: Framework, control approach and implementation," Journal of Manufacturing Systems, vol. 58, 2021, doi: 10.1016/j.jmsy.2020.07.013.

[22] V. Tihanyi et al., "Towards cooperative perception services for its: Digital twin in the automotive edge cloud," Energies, vol. 14, no. 18, 2021, doi: 10.3390/en14185930.

[23] M. Mohammadi, M. Rashidi, V. Mousavi, A. Karami, Y. Yu, and B. Samali, "Quality evaluation of digital twins generated based on uav photogrammetry and tls: Bridge case study," Remote Sensing, vol. 13, no. 17, 2021, doi: 10.3390/rs13173499.

[24] D. A. Howard, Z. Ma, C. Veje, A. Clausen, J. M. Aaslyng, and B. N. Jørgensen, "Greenhouse industry 4.0 – digital twin technology for commercial greenhouses," Energy Informatics, vol. 4, 2021, doi: 10.1186/s42162-021-00161-9.

[25] H. Yu, S. Han, D. Yang, Z. Wang, and W. Feng, "Job Shop Scheduling Based on Digital Twin Technology: A Survey and an Intelligent Platform," Complexity, vol. 2021, 2021, doi: 10.1155/2021/8823273.

[26] A. K. Ghosh, A. S. Ullah, R. Teti, and A. Kubo, "Developing sensor signal-based digital twins for intelligent machine tools," Journal of Industrial Information Integration, vol. 24, 2021, doi: 10.1016/j.jii.2021.100242.

[27] W. Tärneberg, M. Gunnarsson, M. Kihl, and C. Gehrmann, "Demonstration: A cloud-native digital twin with adaptive cloud-based control and intrusion detection," Electronic Communications of the EASST, vol. 80, 2021, doi: 10.14279/tuj.eceasst.80.1133.